

⑫ 公開特許公報(A) 平3-175065

⑤Int.Cl.⁵

識別記号

庁内整理番号

⑬公開 平成3年(1991)7月30日

B 41 J 5/30

Z 8907-2C

29/38

Z 8804-2C

G 06 F 3/12

B 8323-5B

審査請求 未請求 請求項の数 7 (全52頁)

⑭発明の名称 データ・バッファ装置

⑮特 願 平1-315066

⑯出 願 平1(1989)12月4日

⑰発明者 佐 藤 誠 東京都大田区下丸子3丁目30番2号 キヤノン株式会社内
 ⑱発明者 温 泉 隆 広 東京都大田区下丸子3丁目30番2号 キヤノン株式会社内
 ⑲出願人 キヤノン株式会社 東京都大田区下丸子3丁目30番2号
 ⑳代理人 弁理士 丸島 儀一 外1名

明 細 書

1. 発明の名称

データ・バッファ装置

2. 特許請求の範囲

(1) データ送出装置からブロック単位に送出されたデータを受信する受信手段と、

ブロック毎にブロック識別データであるブロック・インデックスを対応させて受信したデータを蓄積する蓄積手段と、

1つのブロックに含まれる全データが蓄積されたときに、そのブロックを登録する登録手段と、

登録したブロックのデータをデータ受入装置に送信する送信手段と、

データの受信動作を中断する受信中断手段と、

中断された受信動作を再開する受信再開手段とを含むことを特徴とするデータ・バッファ装置。

(2) 請求項第1項記載のデータ・バッファ装置において、

受信動作の中断状態にあるときに、その旨を表示する手段を備えることを特徴とするデータ・

バッファ装置。

(3) 請求項第1項又は第2項記載のデータ・バッファ装置において、

登録手段は受信動作の中断されたブロックを1つのブロックとして登録可能であることを特徴とするデータ・バッファ装置。

(4) 請求項第1項～第3項のいずれかに記載のデータ・バッファ装置において、

登録されているブロックに対応する前記ブロック・インデックスを表示する手段を備えることを特徴とするデータ・バッファ装置。

(5) 請求項第1項～第4項のいずれかに記載のデータ・バッファ装置において、

受信動作の中断されたブロックのデータを消去する消去手段を備えることを特徴とするデータ・バッファ装置。

(6) 請求項第5項記載のデータ・バッファ装置において、

受信動作の中断状態にあるときに、前記受信再開手段、登録手段及び消去手段のうちのいずれ

か1つを選択して能動化するための選択手段を備えることを特徴とするデータ・バッファ装置。

(7) 請求項第1項～第6項のいずれかに記載のデータ・バッファ装置において、

データ送出装置のコネクタと直接嵌合可能であり、該コネクタと嵌合した状態でデータ送出装置からのデータを受信するための第1コネクタと、

データ受入装置のコネクタと直接嵌合可能であり、該コネクタと嵌合した状態でデータ送出装置へデータを送信するための第2コネクタとを備えることを特徴とするデータ・バッファ装置。

(以下余白)

バイト・データを受信する用意ができていることを示す。またプリンタから送出される制御信号線BUSYは“High”状態(アサート状態)の時バイト・データを受信が不可能であることを示し、逆に“Low”状態(ネゲート状態)の時バイト・データを受信が可能であることを示す。SLCTはプリンタが送出する制御信号線で“High”状態(アサート状態)の時プリンタがセレクト状態であることを示す。—ERRORはプリンタが送出する制御信号線で“Low”状態(アサート状態)の時プリンタがエラー状態であることを示す。またPEもプリンタから送出される制御信号線で前記—ERROR信号線がアサート状態の時にこの信号線が“High”状態(アサート状態)であると、プリンタが紙無し状態であることを表わしている。—INITはコンピュータからプリンタへ送られる制御信号線で“Low”状態(アサート状態)にするとプリンタは初期状態にセットされる。—AUTOFEEDXTはコンピュータからプリンタへ送出される制御信号線で“Low”状態(アサ

3. 発明の詳細な説明

(産業上の利用分野)

本発明はたとえばコンピュータ等のデータ送出装置と、たとえばプリンタ等のデータ受入装置との間で大量のデータを送受信する際、送受信を効率よく行う目的で両者の間に設置されるデータ・バッファ装置に関するものである。

(従来の技術)

一般的なコンピュータとプリンタ間のデータ送受のための信号線は第58図に示すようになっている。図中、DATA0～DATA7はコンピュータからプリンタに送られるバイト・データ(パラレル・データ)を構成する各ビットの情報を乗せる信号線で、同じくコンピュータからプリンタに送られる制御信号線である。—STROBEは“Low”状態(アサート状態)の時前記DATA0～DATA7信号線に有効なデータが存在することを表わす。—ACKNLGはプリンタからコンピュータに送られる制御信号線で、“Low”状態(アサート状態)の時バイト・データの受信処理が終了し、次の

ト状態)にするとプリンタはキャリッジ・リターン・コードの受信で自動的に1行改行動作を行う。ただしこの信号線—AUTOFEEDXTはプリンタの初期状態においてのみモニタされる。同様にコンピュータから送出される制御信号線—SLCTINもプリンタの初期状態においてのみモニタされ、“Low”状態(アサート状態)であるとプリンタはDC1またはDC3制御コードを無視する。

次に前記第58図に示したコンピュータとプリンタの信号線を用いた一般的なプリント出力データの送受のフローを説明する。第59図は主要信号線のタイミング・チャートである。コンピュータ側におけるプリント出力データの送信動作フローは第60図に示すように、まずステップS10-1で—STROBE信号線を“High”状態にして現在送信動作中で無いことを示した後、ステップS10-2およびステップS10-3で—AUTOFEEDXT信号線および—SLCTIN信号線を“High”または“Low”状態に設定して、ステップS10-4およびステップS10-5で—INIT信号線に“Low”

パルスを送出しプリンタを初期化する。実際のバイト・データの送信動作に際してはステップS10-6でプリンタがセレクト状態であること、ステップS10-7でプリンタがエラー状態でないこと、さらにステップS10-8でプリンタが受信可能状態であることをあらかじめ確認した後、ステップS10-9でプリント出力データを構成するバイト・データをDATA0～DATA7信号線に送出する。そしてステップS10-10およびステップS10-11で-STROBE信号線に“Low”パルスを送出してプリンタにバイト・データを送信したことを知らせる。コンピュータはその後ステップS10-12で-ACKNLG信号線をモニタして“Low”状態（アサート状態）になったならばひとつのバイト・データの送信動作の終了として、S10-13でプリント出力データを構成するすべてのバイト・データを送信し終わったかを判定し、プリント出力データが終了していなければステップS10-6に戻り、再びバイト・データの送信動作を繰り返す。

一方プリンタ側におけるプリント出力データの

プリンタが受信可能状態であることを示した後、ステップS11-12においてコンピュータから送出される-STROBE信号線をモニタし“Low”状態（アサート状態）になったならばまずステップS11-13でBUSY信号線を“High”状態（アサート状態）にしてコンピュータに現在プリンタは受信処理中で次のバイト・データの受信は不可能であることを知らせた後、ステップS11-14でバイト・データをDATA0～DATA7信号線から読み取り、エントリ動作あるいは実際の印字動作を行う。その後ステップS11-15でエラーが発生しなければ再びステップS11-9に戻り次のバイト・データの受信動作を再開する。エラーが発生した時はステップS11-16で-ERROR信号線を“Low”状態（アサート状態）にして、さらにステップS11-17で紙無し状態を検出した場合は加えてステップS11-18でPE信号線を“High”状態（アサート状態）にした後受信動作を終了する。

従来のプリンタ・バツファ装置は第62図に示されるような使用形態をとっていた。図中、100は

受信動作のフローは第61図のようになっており、まずステップS11-1で-INIT信号が“Low”状態（アサート状態）になるのを待ち、“Low”状態になったら、ステップS11-2およびステップS11-3で-AUTOFEEDXT信号線と-SLCTIN信号線の状態をそれぞれセーブする。次にステップS11-4で-ACKNLG信号線を“High”状態（ネゲート状態）に、ステップS11-5でSLCT信号線を“High”状態（アサート状態）に、ステップS11-6でBUSY信号線を“High”状態（アサート状態）に、ステップS11-7でPE信号線を“Low”状態（ネゲート状態）に、ステップS11-8で-ERROR信号線を“High”状態（ネゲート状態）に設定し、コンピュータに対して現在プリンタはセレクト状態で、エラー状態では無いが受信は不可能であることを通知する。実際のバイト・データの受信動作に際してはあらかじめステップS11-9からステップS11-11にかけて-ACKNLG信号線に“Low”パルスを送出しつつ、BUSYを“Low”状態（ネゲート状態）にしてプ

データ送出装置であるコンピュータ、101はプリンタ・バツファ装置本体、102はデータ受入装置であるプリンタ、103はコンピュータとプリンタバツファ装置を接続するケーブル、104はプリンタとプリンタ・バツファ装置を接続するケーブル、105は商用AC電源、106はAC電源ケーブルである。

すなわち従来のプリンタ・バツファ装置101は、データの送受信を行うためにデータ送出装置であるコンピュータ100、およびデータ受入装置であるプリンタ102との間に専用の信号ケーブル103および104を用いて接続を行い、また商用AC電力の供給をうけるために専用のAC電源ケーブル106を使用していた。

そしてこのように接続した後、前記データ送出装置であるコンピュータ100から専用ケーブル103を介してプリント出力データが出力されるとプリンタ・バツファ装置102はこのデータを受信し、一旦、内部バツファに蓄積した後専用ケーブル104を介してデータ受入装置であるプリンタ102に

送出していた。

この際のプリント出力データの送受信方法は、前記の一般的なコンピュータとプリンタ間の送受信方法と全く同様である。すなわち、データ送出装置であるコンピュータからプリント出力データを受信する際にはプリンタ・バッファ装置は前記の一般的なプリンタの受信動作をエミュレートし、またデータ受入装置であるプリンタへプリント出力データを送信する際には、プリンタ・バッファ装置は前記の一般的なコンピュータの送信動作をエミュレートしている。

但しここで一般的なプリンタ動作をエミュレートする際、プリンタの“紙無し”はプリンタ・バッファ装置の内部バッファ残容量によりエミュレートされている。すなわちプリンタ・バッファ装置の内部バッファ残容量が0のとき“紙無し”となっている。

そして実際には従来のプリンタ・バッファ装置は上記のプリント出力データの送受信を同時に行っている。その処理の手順は第63図に示されるようなものである。すなわち、まずステップS63-1

S63-8へ移行しコンピュータから受信し蓄積したプリント出力データを、先頭からファースト・イン・ファースト・アウトの方式でプリンタへ送信する。ここでデータの送信は前記の一般のコンピュータの送信動作をエミュレートすることにより行われる。次に制御はステップS63-9へ移行しプリンタへの送信が終了したプリント出力データを内部バッファから即時消去し内部バッファを整理することにより次の受信データのために内部バッファを解放する。そしてこの後制御は再びステップS63-1へ戻り上記処理を繰り返す。

以上のように従来のプリンタ・バッファ装置はプリント出力データの送受信を行っていた。

(以下余白)

で内部バッファ残容量の判定を行う。残容量=0の場合、制御をステップS63-2へ移行し、PE信号線を“High”にし、その後ステップS63-6へ移行する。一方、残容量≠0の場合、ステップS63-3でPE信号線を“Low”にし次にステップS63-4でコンピュータからデータが送信されてきたかどうか判定を行う。ここでデータが送信されてきた場合にはステップS63-5でデータを受信し内部バッファに蓄積しその後ステップS63-6へ移行する。ここでデータの受信は前記の一般のプリンタの受信動作をエミュレートすることにより行われる。

次にステップS63-6ではプリンタが受信可能状態であるかどうか判定を行う。ここでプリンタが受信可能状態でない場合、制御は即時ステップS63-1へ戻る。一方、プリンタが受信可能状態である場合、制御はステップS63-7へ移行し、内部バッファにデータがあるか否か判定を行う。ここでデータがない場合制御はステップS63-4へ戻る。一方データがある場合制御はステップ

[発明が解決しようとしている課題]

しかし、従来のプリンタ・バッファ装置はデータ送出装置であるコンピュータから送信されるプリント出力データを受信および蓄積中、データ受入装置であるプリンタが受信可能である限り、即時蓄積されているプリント出力データを先頭からファースト・イン・ファースト・アウトの方式でプリンタに送信してしまう。このため、プリンタ・バッファ装置においてコンピュータから送信されるプリント出力データの受信動作を中断させるためにはプリンタを“オフライン”状態にする他無く、実際プリンタが設置されている場所まで移動しなければならず不便であった。

したがって本発明の目的は、容易に受信動作の中断／再開を行うことができ、利便性を向上したデータ・バッファ装置を提供することである。

また従来のプリンタ・バッファ装置はすでにデータ送出装置であるコンピュータから受信および蓄積したプリント出力データと現在コンピュータから受信中であるところのプリント出力データを区

別して管理していないため、現在コンピュータから送信されてプリンタ・バッファ装置によって受信されているプリント出力データが不要であるとユーザが判断した場合でもプリンタ・バッファ装置の蓄積手段から不要なプリント出力データのみを消去する手段が無く、どうしても消去したい場合はプリンタ・バッファ装置をリセットして不要なプリント出力データも含めて蓄積手段に蓄積されたすべてのプリント出力データを消去する他手段は無く大変不便であった。

したがって本発明の他の目的は、蓄積手段内の不必要なデータだけを消去することができ、利便性を向上したデータ・バッファ装置を提供することである。

〔課題を解決するための手段及び作用〕

上記目的を達成するために、本発明に従うデータ・バッファ装置は、データ送出装置からブロック単位に送出されたデータを受信する受信手段と、ブロック毎にブロック識別データであるブロック・インデックスを対応させて受信したデータを蓄積

能になる他、現在データ・バッファ装置の受信動作が中断状態であることを容易に確認することが可能になる。

また本発明に従うデータ・バッファ装置は受信動作の中断されたブロックのデータを消去する消去手段を備えており、受信動作の中断状態にあるときに、前記受信再開手段、登録手段及び消去手段のうちのいずれか1つを選択して能動化するための選択手段を備えることを特徴とする。

したがって本発明に従えば、受信動作の中断状態において現在受信中であったデータを含むブロックを蓄積手段から消去する消去手段、受信動作の中断状態において現在受信中であったデータを含むブロックをひとつのブロックとして登録可能な登録手段および受信動作の中断状態において受信動作を再開する受信再開手段を設け、さらに前記消去手段、登録手段及び受信再開手段のいずれかの動作を選択する選択手段を設けることによって、蓄積手段に蓄積されたすべてのデータを消去することなく不要となる現在受信中のデータ

する蓄積手段と、1つのブロックに含まれる全データが蓄積されたときに、そのブロックを登録する登録手段と、登録したブロックのデータをデータ受入装置に送信する送信手段と、データの受信動作を中断する受信中断手段と、中断された受信動作を再開する受信再開手段とを含むことを特徴とする。このようなデータ・バッファ装置において、受信動作の中断状態にあるときに、その旨を表示する手段を備えていてもよい。また前記登録手段は受信動作の中断されたブロックを1つのブロックとして登録可能であってもよく、登録されているブロックに対応する前記ブロック・インデックスを表示する手段を備えていてもよい。

本発明に従えば、データ送出装置から送信されるデータの受信動作を中断する受信中断手段、現在受信動作が中断状態であることを表示する手段、さらに受信動作を再開する受信再開手段を設けることによって、ユーザはデータ送出装置が設置されている場所でデータ・バッファ装置のデータの受信動作の中断／再開指示を容易に行うことが可

を含むブロックのみを消去することが可能になる。

本発明は、データ送出装置のコネクタと直接嵌合可能であり、該コネクタと嵌合した状態でデータ送出装置からのデータを受信するための第1コネクタと、

データ受入装置のコネクタと直接嵌合可能であり、該コネクタと嵌合した状態でデータ送出装置へデータを送信するための第2コネクタとを備えるデータ・バッファ装置に、特に好適に実施することができる。

(以下余白)

〔実施例〕

(I) 第1図から第57図は本発明の実施例を表わしている。

第1図は本発明の実施例であるところのプリンタ・バッファの外観図であり、第2図はその平面図である。第1図および第2図において1はプリンタ・バッファ本体、2は対プリンタ・ポート・コネクタ、3は対コンピュータ・ポート・コネクタ、4はリセット・ボタン、5はクリア・ボタン、6は受信開始／終了ボタン、7は受信中断ボタン、8は送信開始／終了ボタン、9は送信中断ボタン、10はブロック・インデックス・ボタンで全部で20個ありそれぞれ1から20まで番号が付けられている。11は全ブロック選択ボタンである。12は電源インジケータ・ランプ、13は受信中表示ランプ、14は送信中表示ランプ、15はバッファ空表示ランプ、16はバッファ・フル表示ランプ、17は残バッファ／未送信データ容量表示ランプで0から100までの整数を表示できる。18はブロック・インデックス表示ランプで全部で20個ありそれぞれ1から20

タ及び制御信号線のリード／ライトを前記CPU22から可能にしていると同時に－INIT－C信号がアクティブになったことをリセット割込み信号線28を用いて前記CPU22に通知する機能を有する。28は前述したようにリセット割込み信号線である。

29は入力パネルであり、ここには前述したリセット・ボタン4、クリア・ボタン5、受信開始／終了ボタン6、受信中断ボタン7、送信開始／終了ボタン8、送信中断ボタン9、ブロック・インデックス・ボタン10、および全ブロック選択ボタン11が配置されている。30は入力ポートであり、前記入力パネル29上の前記各ボタンの操作状況（押下されているか否か）の読み出しを前記CPU22から可能にするとともに、前記リセット・ボタン4が押下された時には前記CPU22に対してリセット割込み信号線28を用いて通知する機能を有する。31は表示パネルであり、ここには前述した受信中表示ランプ13、送信中表示ランプ14、バッファ空表示ランプ15、バッファ・フル表示ランプ16、

までの番号が付けられている。19は電源スイッチである。

第3図は本発明の実施例であるところのプリンタ・バッファのブロック図を表わしており、図中21は電源部であり電池およびレギュレータ等で構成される。19は電源部21に接続されシステムへの電源供給を制御する電源スイッチ、12は電源部21に接続され電源供給のオン／オフを表示する電源インジケータ・ランプ、22は該プリンタ・バッファの動作制御の中核であるところのCPU、23は前記CPU22に接続され後述する各機能ブロックとの間でデータを授受するためのバス、24は前記CPU22上で動く動作制御プログラムが格納されているROM、25は前記CPU22がある一定時間の経過したことを認知し得る機能を有するタイマ、26は対プリンタ・ポートであり、前記対プリンタ・ポート・コネクタ2に対するデータ及び制御信号線のリード／ライトを前記CPU22から可能にしている。27は対コンピュータ・ポートであり、前記対コンピュータ・ポート・コネクタ3に対するデー

残バッファ／未送信データ容量表示ランプ17、およびブロック・インデックス表示ランプ18が配置されている。32は出力ポートであり、前記表示パネル31上の前記各ランプへの点燈設定（点燈／点滅／消燈）の書き込みを前記CPU22から可能にしている。

33は主メモリで後述する制御パラメータ・テーブルの格納場所として一部使用する他はほとんどプリント出力データの受信バッファとして用いられる。

第4図は本発明の実施例であるところのプリンタ・バッファ1の使用形態を表わした図である。

該図左側はデータ送出装置であるところのコンピュータからのプリント出力データの受信動作の使用形態を表わしており、図中40はプリント出力データを送信するコンピュータであり、41は前記コンピュータ40に装備されているプリンタ・ポート・コネクタであり、プリンタ・バッファ1は該コネクタ41に前記対コンピュータ・ポート・コネクタ3を介して接続されている。このように接続され

た後、プリンタ・バッファ1は前記コンピュータ40から出力されたプリント出力データを受信する。

一方第4図の右側はデータ受入装置であるところのプリンタへのプリント出力データの送信動作の使用形態を表わしており、図中42はプリント出力データを受信し実際プリント・アウトを行うプリンタであり、43は前記プリンタ42に装備されているコンピュータ・ポート・コネクタであり、プリンタ・バッファ1は該コネクタ43に前記対プリンタ・ポート2を介して接続されている。このように接続された後、すでに前記コンピュータ40から受信され、プリンタ・バッファ1に蓄積されているプリント出力データをプリンタ・バッファ1から前記プリンタ42へ送信して結果的にプリンタ42からプリント・アウトを得る。

第5図は本発明の実施例であるところのプリンタ・バッファの前記対プリンタ・ポート2及び対コンピュータ・ポート3それぞれにおけるデータおよび制御信号線の取り扱いを表わしたものである。

この図を見るとわかるように各々のポートに

したものである。

図中、上段に示したブロック終了コードによるデータ・ブロッキングはコンピュータから送出されるプリント出力データが各々あらかじめ決められたブロック終了コードによって終結している場合であり、このような場合プリンタ・バッファはそのブロック終了コードを検知してひとつのデータ・ブロックの終結とみなす。すなわちコンピュータから送出されるプリント出力データとプリンタ・バッファ内で管理されるデータ・ブロックは完全に1対1で対応づけされる。

図中、中段はブロック終了操作によるデータ・ブロッキングを示しており、この場合コンピュータから送出されるプリント出力データは前述のブロック終了コードによって終結しておらず、ユーザー（オペレータ）のあらかじめ決められた手順に従ったブロック終了操作によってプリンタ・バッファはひとつのデータ・ブロックを終結する。ここに示してある例のようにプリンタ・バッファ内で管理されるデータ・ブロックは複数のプリント

入力する信号はすべて非接続時（コンピュータあるいはプリンタに接続されていない時）に“High”状態となるように電氣的にプル・アップされている。また-AUTOFEEDXT__Cおよび-SLCTIN__Cの対コンピュータ・ポートの2つの制御信号線については、イニシャライズ時にのみモニタされその情報が主メモリ33に保存される。

対コンピュータ・ポートの制御信号線-INIT__Cはリセット信号線として扱われ、“Low”状態がコンピュータから入力されると前記リセット割込み信号線28がアサートされCPU22に通知された後、イニシャライズ動作に移行する。

出力信号線のうち対コンピュータ・ポートの制御信号線SLCT__Cは常に“High”状態が出力されるようになっている。また対プリンタ・ポートの-INIT__P、-AUTOFEEDXT__Pそして-SLCTIN__Pの3つの制御信号線はすべて送信動作開始時にのみ出力される。

第6図は本発明の実施例であるところのプリンタ・バッファにおけるデータ・ブロックの概念を表わ

出力データによって構成される場合があり、同図上段の場合と違ってプリント出力データとデータ・ブロックは完全に1対1で対応づけされない。

図中、下段は受信バッファ・フル時におけるデータ・ブロッキングを示しており、この場合プリンタ・バッファはプリント出力データ受信中にその格納メモリ領域であるところの受信バッファが満杯になったことを検知して、その後ユーザーのあらかじめ決められた手順に従ったブロック中断あるいは終了操作によってデータ・ブロックの終結動作を行う。

いずれの場合にせよ、データ・ブロックは該プリンタ・バッファで管理されるプリント出力データの単位であることには変わらない。

第7図は本発明の実施例であるところのプリンタ・バッファにおける前記主メモリ33の内容を表わしたものである。

図中、割込みベクタ・テーブルはCPUの例外処理及び外部割込みサービス・ルーチンのポイントが格納されている領域、制御パラメータ・テ-

ブルは該プリンタ・バッファの動作制御に必要な各種パラメータが格納されている領域、スタック領域は文字どおり前記CPU22にとってのスタック領域、そして受信バッファは該プリンタ・バッファがコンピュータから受信するデータ・ブロックの格納用に使用できる領域である。便宜的に、以後そのスタート・アドレスをMEMSTAおよびそのサイズ(バイト数)をMEMMAXとする。

第8図は前記主メモリ33上に設けられた前記制御パラメータ・テーブルの構成の概要を表わしたものである。図を見ると明らかなように該制御パラメータ・テーブルは環境テーブル、ブロック登録テーブル、受信順テーブル、そして送信順テーブルの4つの部分によって構成されている。前記環境テーブルは当該プリンタ・バッファの動作制御における動作環境を表わすパラメータによって構成されるテーブルであり、前記ブロック登録テーブルはコンピュータから受信され登録されたデータ・ブロック個々のパラメータによって構成されるテーブルであり、前記受信順ブロック

はコンピュータから受信され登録されたデータ・ブロックの受信順序を表わすパラメータによって構成されるテーブルであり、前記送信順ブロックはプリンタへ送信されるデータ・ブロックの送信順序を表わすパラメータによって構成されるテーブルである。

第9図は第8図で示した前記環境テーブルの構成を詳しく表わしたものである。

図中、RBLOCKIは次にコンピュータから受信されるデータ・ブロックに与えるブロック・インデックスあるいは現在コンピュータから受信中のデータ・ブロックのブロック・インデックスが格納されている1バイトの領域で、1から20のうちいずれかの値をとり得る。イニシャライズ時はデフォルト値として“1”が設定されている。

TBLOCKIは次にプリンタへ送信されるデータ・ブロックのブロック・インデックスあるいは現在プリンタへ送信中のデータ・ブロックのブロック・インデックスが格納されている1バイトの領域で、1から20のうちいずれかの値をとり得る。イニ

シャライズ時はデフォルト値として“1”が設定されている。

RBLOCKSは現在すでにコンピュータから受信され、登録されているデータ・ブロックの総数が格納されている1バイトの領域で、0から20の間の値をとり得る。この値が0の時は登録されているデータ・ブロックが皆無であることを表わし、20の時は登録されているデータ・ブロックの総数が最大ブロック登録数に達していることを表わしている。イニシャライズ時は“0”にクリアされている。TBLOCKSは現在プリンタへ送信中のデータ・ブロックも含めて送信動作開始後すでにプリンタへ送信したデータ・ブロックの総数が格納されている1バイトの領域で、0から20の間の値をとり得るが、前記登録済ブロック数RBLOCKSの値を超えることは無い。すなわち、

$$0 \leq TBLOCKS \leq RBLOCKS \leq 20$$

の関係が成り立つ。この値が0の時は該プリンタ・バッファのプリンタへの送信動作が行われていないことを表わし、RBLOCKSの値に一致している

時は登録されているすべてのデータ・ブロックをプリンタへ送信したかあるいは最後のひとつのデータ・ブロックをプリンタへ送信中であるかのいずれかの動作状態であることを表わしている。イニシャライズ時は“0”にクリアされている。

UMEMは当該プリンタ・バッファがコンピュータから受信するデータ・ブロックの格納用に使用できる領域すなわち第7図で示した前記受信バッファのサイズ(総バイト数)が格納されている3バイト(24ビット)長の領域で、イニシャライズ時に値MEMMAXが設定されている。

RREMは前記受信バッファの総バイト数から現在すでにコンピュータから受信したデータ・ブロックが格納されている部分の総バイト数を差し引いた残りのメモリ量(バイト数)が格納されている3バイト(24ビット)長の領域で0から前記受信バッファのサイズUMEMの値(MEMMAX)の間の値をとり得る。この値がMEMMAXの時は、残受信バッファ量が最大値すなわち“バッファ空”状態であることを表わし、0の時は残受信バッファ

量が皆無すなわち“バッファ・フル”状態であることを表わしている。イニシャライズ時は値MEMMAXが設定されている。

TBLOCKSIZは現在プリンタへ送信中のデータ・ブロックについてすでにプリンタへ送信したバイト・データの総数(バイト数)が格納されている3バイト(24ビット)長の領域で、0から当該データ・ブロックのサイズ(バイト数)までの値をとり得る。この値が0の時は当該データ・ブロックのプリンタへの送信動作が行われていないことを表わし、当該データ・ブロックのサイズに一致している時は該データ・ブロックのプリンタへの送信動作が終了したことを表わしている。イニシャライズ時は0にクリアされている。

TSIZはプリンタへ送信すべきすべてのデータ・ブロックのサイズ(バイト数)の総数が格納されている3バイト(24ビット)長の領域である。この値が0の時はプリンタへ送信すべきデータ・ブロックが存在しないことを表わしている。イニシャライズ時は“0”にクリアされている。

“Low”)が、ビット__1にあたるAFDXTには前記制御信号線-AUTOFEEDXT_Cの状態(“High”または“Low”)がそれぞれ格納されている。またビット__7からビット__2にあたる5ビットは定義されておらず未使用である。

第11図は第8図で示した前記ブロック登録テーブルの構成を詳しく表わしたものである。

図中、BLOCKADR[i]($1 \leq i \leq 20$)は、ブロック・インデックスがiとして登録された(またはコンピュータから受信中の)データ・ブロックの受信バッファにおけるスタート・アドレスが格納されている3バイト(24ビット)長の領域であり、BLOCKSIZ[i]($1 \leq i \leq 20$)は該データ・ブロックのサイズ(バイト数)が格納されている3バイト(24ビット)長の領域である。イニシャライズ時BLOCKADR[1]にはデフォルト値として前記受信バッファのスタート・アドレスMEMSTAが設定されBLOCKSIZ[1]は0にクリアされている。

第12図は第8図で示した前記受信順テーブルの

TREMは前記プリンタへ送信すべきすべてのデータ・ブロックの総バイト数TSIZから現在すでにプリンタへ送信したバイト・データの総バイト数を差し引いた残りのメモリ量(バイト数)が格納されている3バイト(24ビット)長の領域で0から前記送信すべきすべてのバイト・データ数TSIZの間の値をとり得る。この値がTSIZの値と一致するときは、送信動作開始後プリンタへ現在まだ1バイトのデータも送信していないことを表わし、0の時は送信すべきすべてのデータ・ブロックの送信動作が終了したことを表わしている。イニシャライズ時は“0”にクリアされている。

STATEは対コンピュータ・ポートの制御信号線のイニシャライズ時における状態が格納されている1バイト(8ビット)の領域である。

第10図は前記対コンピュータ・ポートの制御信号線のイニシャル状態STATEのビット構成を表わしたものである。

図中ビット__0にあたるSLINには前記制御信号線、-SLCTIN_Cの状態(“High”または

内容を詳しく表わしたものである。

図中、RBLOCKSEQ[m]($1 \leq m \leq 20$)は受信動作開始後m番目に受信登録されたデータ・ブロックのブロック・インデックスの値がそれぞれ格納されている各々1バイトの領域で1から20あるいは0の値をとり得る。例えばRBLOCKSEQ[3]=4は3番目に受信登録されたデータ・ブロックのブロック・インデックスが4であることを表わしている。値が0のときは登録されたデータ・ブロックが存在しないことを表わす。

このテーブルは、順番を表わすという性質上RBLOCKSEQ[1]から順に見た場合0の値を持つ領域以降に有効なデータは存在しないように管理されている。すなわちRBLOCKSEQ[N]≠0かつRBLOCKSEQ[N+1]=0ならばそれ以降のすべての領域についてRBLOCKSEQ[N+2]=RBLOCKSEQ[N+3]=…=RBLOCKSEQ[20]=0となるように設定されている。

この場合登録されているデータ・ブロックの数(前記環境テーブルの中のRBLOCKSの値)は

Nである。

また0以外の値が複数個重複して設定されることはない。イニシャル時はすべての領域が0にクリアされている。すなわち

$$TBLOCKSEQ[m] = 0 \quad (1 \leq m \leq 20)$$

となっている。

第13図は第8図で示した前記送信順テーブルの内容を詳しく表わしたものである。

図中、 $TBLOCKSEQ[m]$ ($1 \leq m \leq 20$) は送信動作開始後 m 番目に送信されるデータ・ブロックのブロック・インデックスの値がそれぞれ格納されている各々1バイトの領域で1から20あるいは0の値を取り得る。例えば $TBLOCKSEQ[3] = 4$ は3番目に送信されるデータ・ブロックのブロック・インデックスが4であることを表わしている。値が0のときは送信されるブロックが存在しないことを表わす。

このテーブルは、順番を表わすという性質上 $TBLOCKSEQ[1]$ から順に見た場合0の値を持つ領域以降に有効なデータは存在しないように

投入されたり、各動作中にリセット・ボタン4が押下されたり、あるいは受信動作中に対コンピュータ・ポートの制御信号線 $-INIT_C$ がアクティブになると当該プリンタ・バッファの動作制御は速やかにステップS1に移行しイニシャライズ動作を行う。ステップS1でのイニシャライズ動作が終了するとステップS2に移行してアイドル状態になり動作指示を待つことになる。ここで前記受信開始/終了ボタン5が押下されるとステップS3へ移行し受信動作を開始する。またステップS2で前記送信開始/終了ボタン6が押下されるとステップS4へ移行し送信動作を開始する。またステップS2で前記クリア・ボタン5が押下されるとステップS5へ移行しデータ・ブロックの選択クリア動作を開始する。ステップS3, S4およびS5のいずれの動作においてもその動作が終了した場合は再びステップS2のアイドル状態に戻り次の動作指示を待つことになる。

第15図は第14図で示したステップS1のイニシャライズ動作での動作制御の概略を表わすフロー

管理されている。すなわち $TBLOCKSEQ[N] \neq 0$ かつ $TBLOCKSEQ[N+1] = 0$ ならばそれ以降のすべての領域について $TBLOCKSEQ[N+2] = TBLOCKSEQ[N+3] = \dots = TBLOCKSEQ[20] = 0$ となるように設定されている。

この場合送信されるデータ・ブロックの数(前記環境テーブルの中の $TBLOCKS$ の値)はNである。

また0以外の値が複数個重複して設定されることはない。イニシャル時はすべての領域が0にクリアされている。すなわち

$$TBLOCKSEQ[m] = 0 \quad (1 \leq m \leq 20)$$

となっている。

(II) つぎに以上説明してきた本発明の実施例であるところのプリンタ・バッファ動作制御フローを第14図～第57図を参照して詳しく説明する。

第14図は本発明の実施例であるところのプリンタ・バッファの動作制御の概略を表わしたものである。

動作制御の概略は第14図に示すように、電源が

チャートである。上記イニシャライズ動作は第15図に示すように、まずステップS1-1で対コンピュータ・ポート27および対プリンタ・ポート26それぞれの出力側制御信号線の出力設定を行う。ここで設定する出力側制御信号線とその状態("High"または"Low")は以下のようになっている。

(1) 対コンピュータ・ポート制御信号線

$-ACKNLG_C \leftarrow "High"$

$BUSY_C \leftarrow "High"$

$PE_C \leftarrow "Low"$

$-ERROR_C \leftarrow "High"$

(2) 対プリンタ・ポート制御信号線

$-STROBE_P \leftarrow "High"$

$-INIT_P \leftarrow "High"$

$-AUTOFEEDXT_P \leftarrow "High"$

$-SLCTIN_P \leftarrow "High"$

ここで(1)対コンピュータ・ポート制御信号線の設定では、コンピュータに対して当該プリンタ・バッファは紙無し状態ではないが、 $BUSY_C$ を

“High”としてビジー状態であることを示している。一方、(2) 対プリンタ・ポート制御信号線の設定ではプリンタに対して当該プリンタ・バッファはプリント出力データを送出していないことを示している。

つぎにステップS1-2へ進み対コンピュータ・ポート27の入力側制御信号線-AUTOFEEDXT_Cおよび-SLCTIN_Cの状態(“High”または“Low”)を読み取りその情報を前記環境テーブル(第9図)の中のSTATEに格納する。すなわち

```
AFDXT ← -AUTOFEEDXT_C
SLIN ← -SLIN_C
```

という操作を行う。

つぎにステップS1-3へ進み表示パネル31上のそれぞれの表示ランプの設定を行う。ここで設定する表示ランプとその表示状態(“点燈”、“点滅”または“消燈”)は以下のようにになっている。

```
受信中表示ランプ13 ← “消燈”
送信中表示ランプ14 ← “消燈”
```

```
TREM ← “0”
```

(2) ブロック登録テーブル(第11図)

```
BLOCKADR[1] ← “MEMSTA”
BLOCKADR[2] ~ [20] ← “0”
BLOCKSIZ[1] ~ [20] ← “0”
```

(3) 受信順テーブル(第12図)

```
RBLOCKSEQ[1] ~ [20] ← “0”
```

(4) 送信順テーブル(第13図)

```
TBLOCKSEQ[1] ~ [20] ← “0”
```

ただし環境テーブルの中のSTATEについてはすでにステップS1-2で値が設定されているので、ここでは初期値設定の対象からはずしてある。

以上でステップS1のイニシャライズ動作は終了する。

第16図は第14図で示したステップS2におけるアイドル状態での動作制御のフローチャートである。上記アイドル状態では、第16図に示すように、まずステップS2-1で受信開始/終了ボタン6が押下されたか否かを検知して、押下された場合はステップS3へ移行し受信動作を行う。押下されて

```
バッファ空表示ランプ15 ← “点燈”
バッファ・フル表示ランプ16 ← “消燈”
残バッファ/未送信データ容量表示ランプ17 ← “100%”
ブロック・インデックス表示ランプ18 ← すべて “消燈”
```

上記の表示状態によってユーザーに現在送信、受信いずれの動作も行われていないことを、および受信バッファが空の状態であることを知らせる。

つぎにステップS1-4へ進み制御パラメータ・テーブル(第8図)の初期設定を行う。ここで設定する制御パラメータ・テーブルのパラメータとその設定値はそれぞれ以下のようにになっている。

(1) 環境テーブル(第9図)

```
RBLOCKI ← “1”
TBLOCKI ← “1”
RBLOCKS ← “0”
TBLOCKS ← “0”
UMEM ← “MEMMAX”
RREM ← “MEMMAX”
TBLOCKSIZ ← “0”
TSIZ ← “0”
```

いなかった場合はステップS2-2へ移行し、送信開始/終了ボタン8が押下されたか否かを検知して、押下された場合はステップS4へ移行し送信動作を行う。押下されていなかった場合はステップS2-3へ移行し、クリア・ボタン5が押下されたか否かを検知して、押下された場合はS5へ移行しデータ・ブロックの選択クリア動作を行う。押下されていなかった場合は再びステップS2-1へ戻る。すなわち受信開始/終了ボタン6、送信開始/終了ボタン7あるいはクリア・ボタン5のいずれかのボタンが押下されるのを待つことになる。ただしリセット・ボタン4が押下された場合は割込みシーケンスによって速やかにステップS1のイニシャライズ動作へ移行する。

第17図は第14図で示した受信動作ステップS3での動作制御の概略を表わしたものである。この受信動作ステップS3では、第17図に示すように、まずステップS3-1で受信動作条件判定を行う。

第18図はこの受信動作条件判定ステップS3-1での動作制御のフローチャートである。第18図に

示した詳しいフローを見ればわかるように、ここでは登録済ブロック数と受信バッファの残容量の判定を行う。登録済ブロック数の判定ステップS3-1-1では登録済ブロック数RBLOCKSが最大登録数20未満であること、すなわち

$$RBLOCKS < 20$$

を判定し、さらに受信バッファの残容量の判定ステップS3-1-2では残受信バッファ量が0でないこと、すなわち

$$RREM > 0$$

を判定する。この2つの条件をいずれも満たす場合制御はステップS3-2に移行し、いずれかが満たされなかった場合は受信動作ステップS3を終了する。

第17図に戻って、ステップS3-2では受信ランプ13を点燈させユーザーに受信動作が開始したことを知らせステップS3-3へ進む。

第19図は第17図で示した開始バイト受信ステップS3-3での動作制御のフローチャートである。ステップS3-3ではブロック・インデックスiの

値が受信バッファのサイズUMEMの値と等しい、いわゆる“バッファ空状態”であった場合はステップS3-3-8のバイト・データdの格納によって少なくとも“バッファ空状態”で無くなった訳であるからステップS3-3-11においてバッファ空表示ランプ15が消燈される。ステップS3-3-12では残受信バッファ量RREMから1が減じられる。続くステップS3-3-13では残受信バッファ量RREMの受信バッファのサイズUMEMに対する割合を百分率で残バッファ／未送信データ容量表示ランプ17に表示する。ここで割合の百分率は次の式によって算出する。

$$\text{残バッファ割合} = \text{整数部} \left(\frac{RREM}{UMEM} \times 100 \right)$$

最後にステップS3-3-14で残受信バッファ量RREMが0すなわち“バッファ・フル状態”になったか否かを判定する。バッファ・フル状態に陥った場合はステップS3-4に移行し、そうで無い場合はステップS3-3における開始バイト受信の正常受信としてステップS3-5に移行する。またステップ

データ・ブロックの最初のバイト・データの受信を行う。バイト・データの受信動作は第19図に示すごとく、-ACKNLG_Cのアサート(ステップS3-3-1)→BUSY_Cのネゲート(ステップS3-3-2)→-ACKNLG_Cのネゲート(ステップS3-3-3)→-STROBE_Cのアサート検出(ステップS3-3-5)→BUSY_Cのアサート(ステップS3-3-6)→DATA0_C～DATA7_Cからバイト・データの読み出し(ステップS3-3-7)というシーケンスで行われる。ステップS3-3-7で読み出されたバイト・データdはステップS3-3-8においてメモリ・アドレスBLOCKADR[i]で示される受信バッファ内の領域に格納される。その後ステップS3-3-9においてデータ・ブロックのサイズBLOCKSIZ[i]に1が加えられる。ステップS3-3-10における判定において、ステップS3-3-8で格納したバイト・データdが受信バッファに格納した最初のバイト・データであった場合、つまりステップS3-3-10の時点で残受信バッファ量RREMの

S3-3-4において-STROBE_Cアサート検出ステップS3-3-5の前に受信開始／終了ボタン6が押下された場合、受信動作の中止とみなし、ステップS3-3-15でBUSY_Cをアサートした後、ステップS3-6へ移行し、そこで点燈中の受信中表示ランプ13を消燈して受信動作ステップS3は終了する。

再び第17図を参照して開始バイト受信S3-3において、上述したように正常受信後ステップS3-5に制御が移り、ここでブロック・インデックスiに対応するブロック・インデックス表示ランプ18を点滅させ、ユーザーに現在ブロック・インデックスiのデータ・ブロックの受信中であることを知らせる。その後ステップS3-7へ移行する。

第20図は第17図で示したデータ・ブロック受信ステップS3-7での動作制御のフローチャートである。ステップS3-7ではブロック・インデックスiのデータ・ブロックの受信を行う。バイト・データの受信動作は第20図に示すごとく、-ACKNLG_Cのアサート(ステップS3-7-1)

→BUSY_Cのネゲート(ステップS3-7-2)→
 →ACKNLG_Cのネゲート(ステップS3-7-3)
 →-STROBE_Cのアサート検出(ステップS3-7-6)→BUSY_Cのアサート(ステップS3-7-7)→DATA0_C~DATA7_Cからバイト・データの読み出し(ステップS3-7-8)というシーケンスで行われる。ステップS3-7-8で読み出されたバイト・データdに対してステップS3-7-9においてデータ・ブロックの終了を示すコード“EOF”(=0BH)であるか否かを判別し、“EOF”であった場合には現在受信中のブロック・インデックスiのデータ・ブロックの受信が終了したものとみなし、ステップS3-8へ移行する。“EOF”で無かった場合はステップS3-7-10へ進みステップS3-7-8で読み出されたバイト・データdをメモリ・アドレスBLOCKADR(i)+BLOCKSIZ(i)で示される受信バッファ内の領域に格納する。その後ステップS3-7-11においてデータ・ブロックのサイズBLOCKSIZ(i)に1が加えられ、ステップS3-7-12において

した後ステップS3-11へ移行する。

前記データ・ブロック受信ステップS3-7においてブロック終了コード“EOF”を受信した場合はステップS3-8へ制御が移り、ここで受信したデータ・ブロックの登録を行う。

第21図は第17図で示した受信データ・ブロック登録ステップS3-8での動作制御のフローチャートである。ここでは第21図に示すように最初にステップS3-8-1で登録済ブロック数RBLOCKSに1を加えた後、ステップS3-8-2で受信順テーブルに登録済みのデータ・ブロックのブロック・インデックスiを格納する。その際、受信順テーブルRBLOCKSEQ[1]~[20]における格納位置(1~20)はステップS3-8-1で加算済みの登録済ブロック数RBLOCKSの値であり、格納領域はRBLOCKSEQ[RBLOCKS]で表わすことができる。またブロック・インデックスiはRBLOCKIに格納されている値である。

すなわちここでの操作は

RBLOCKSEQ[RBLOCKS]←RBLOCKI

残受信バッファ量RREMから1が減じられる。続くステップS3-7-13では残受信バッファ量RREMの受信バッファのサイズUMEMに対する割合を百分率で残バッファ/未送信データ容量表示ランプ17に表示する。割合の百分率の算出式は前記ステップS3-3-13で用いたものと同じである。最後にステップS3-7-14で残受信バッファ量RREMが0すなわち“バッファ・フル状態”になったか否かを判定する。バッファ・フル状態に陥った場合はステップS3-4に移行し、そうで無い場合は再びステップS3-7-1に戻り、次のバイト・データの受信を行う。また-STROBE_Cアサート検出ステップS3-7-6の前にステップS3-7-4において受信開始/終了ボタン6が押下された場合、受信動作の中止とみなしステップS3-7-15でBUSY_Cをアサートした後ステップS3-10へ移行し受信中のデータ・ブロックを消去する。同様にステップS3-7-5において受信中断ボタン7が押下された場合、受信動作の中断とみなしステップS3-7-16でBUSY_Cをアサート

で表わすことができる。その後ステップS3-8-3で点滅中のブロック・インデックスiに対応するブロック・インデックス表示ランプ18を点燈させ、ユーザーにブロック・インデックスiのデータ・ブロックが登録されたことを知らせた後ステップS3-9へ移行する。

第22図は第17図で示した次ブロック・インデックス設定ステップS3-9での動作制御のフローチャートである。

ステップS3-9では次にコンピュータから受信されるデータ・ブロックに与えるブロック・インデックスRBLOCKIの設定を行なう。第22図に示すようにまずステップS3-9-1において現在登録済みのデータ・ブロック数RBLOCKSが最大登録数20未満であることを判定し、条件を満たさない場合はステップS3-6へ移行し、速やかに受信動作を終了する。条件を満足した場合はステップS3-9-2へ進み、ここで候補となる未使用のブロック・インデックスxの初期値として“1”を代入する。その後ステップS3-9-3において

ブロック・インデックス x のデータ・ブロックがすでに登録済みか否かをそのサイズであるところの $BLOCKSIZE[x]$ が 0 かどうかによって判定する。0 の場合は、ブロック・インデックス x のデータ・ブロックは未登録であるとみなしステップ $S3-9-5$ へ進みそこでブロック・インデックス x のデータ・ブロックを格納する受信バッファでのスタート・アドレス $BLOCKADR[x]$ を設定する。ここで設定するスタート・アドレスの値は最後にステップ $S3-7$ で受信していたブロック・インデックス i のデータ・ブロックのスタート・アドレス $BLOCKADR[i]$ とそのサイズ $BLOCKSIZE[i]$ の和を用いている。すなわち

$$BLOCKADR[x] \leftarrow BLOCKADR[i] + BLOCKSIZE[i]$$

で表わすことができる。これは次に受信するデータ・ブロックは受信バッファにおいて最後に受信登録されたデータ・ブロックに続けて格納されることを意味する。スタート・アドレスの設定ステップ $S3-9-5$ の後ステップ $S3-9-6$ へ進みいよいよ次に受信するデータ・ブロックに与える

のデータ・ブロックの強制消去を行なう。

第 23 図は第 17 図で示した受信データ・ブロック消去ステップ $S3-10$ での動作制御のフローチャートである。ここではまずステップ $S3-10-1$ で残受信バッファ量 $RREM$ に現在までステップ $S3-7$ で受信していたブロック・インデックス i のデータ・ブロックのサイズ $BLOCKSIZE[i]$ を加えることにより残受信バッファ量 $RREM$ の値をブロック・インデックス i のデータ・ブロックの受信開始時の状態に戻す。つぎにステップ $S3-10-2$ で復元した残受信バッファ量 $RREM$ の受信バッファのサイズ $UMEM$ に対する割合を百分率で残バッファ／未送信データ容量表示ランプ 17 に表示する。割合の百分率の算出式は前記ステップ $S3-3-13$ で用いたものと同じである。続いてステップ $S3-10-3$ で復元した残受信バッファ量 $RREM$ の値が受信バッファのサイズ $UMEM$ の値と等しいいわゆる“バッファ空状態”であるか否かを判定し、バッファ空状態ならばステップ $S3-10-4$ においてバッファ空表示ランプ 15 を点燈する。そしてステップ

ブロック・インデックス $RBLOCKI$ に x が格納される。一方、ステップ $S3-9-3$ の判定において $BLOCKSIZE[x]$ が 0 で無かった場合はブロック・インデックス x のデータ・ブロックは既に登録されている訳であるからステップ $S3-9-4$ へ進み、候補となるブロック・インデックス x に 1 を加えた後再びステップ $S3-9-3$ の判定を繰り返す。この方法において使用されていないブロック・インデックスはステップ $S3-9-1$ における判定で少なくともひとつは存在することがわかっているので、必ず未登録のデータ・ブロックに対するブロック・インデックス x が得られる。以上のようにステップ $S3-9-6$ において新たにコンピュータから受信するデータ・ブロックのブロック・インデックス $RBLOCKI$ が設定されたら再びステップ $S3-3$ に戻り、新たなデータ・ブロックの最初のバイト・データの受信を行なう。

前記データ・ブロック受信ステップ $S3-7$ において受信開始／終了ボタン 6 が押下された場合にはステップ $S3-10$ に制御が移り、ここで、受信中

$S3-10-5$ において現在までステップ $S3-7$ で受信していたブロック・インデックス i のデータ・ブロックのサイズ $BLOCKSIZE[i]$ を 0 にクリアすることにより受信データ・ブロック消去が終了し、ステップ $S3-10-6$ で点滅中のブロック・インデックス i に対応するブロック・インデックス表示ランプ 18 を消燈させ、ユーザーにブロック・インデックス i のデータ・ブロックが消去されたことを知らせた後ステップ $S3-6$ へ進むことにより受信動作が終了する。

前記データ・ブロック受信ステップ $S3-7$ において受信中断ボタン 7 が押下された場合にはステップ $S3-11$ に制御が移り、ここで点燈中の受信中表示ランプ 13 を点滅させユーザーに受信動作が中断中であることを知らせた後ステップ $S3-12$ へ進む。

第 24 図は第 17 図で示した指示待ち $S3-12$ での動作制御のフローチャートである。

ステップ $S3-12$ は受信動作中断時における動作指示待ち状態であり、第 24 図に示すように動作

指示のボタンのいずれかが押下されるまで待機している。受信中断ボタン7が押下された場合（ステップS3-12-1）、制御はステップS3-13に移行し、受信開始／終了ボタン6が押下された場合（ステップS3-12-2）制御はステップS3-14に移行し、クリア・ボタン5が押下された場合（ステップS3-12-3）、制御はステップS3-16に移行する。

ステップS3-12の中断時の動作指示待ち状態において受信中断ボタン7が押下されると受信動作再開（中断解除）として扱い、ステップS3-13へ移行し、ここで点滅中の受信中表示ランプ13を点燈させユーザーに受信動作が再開したことを知らせた後、ステップS3-7へ戻り再びブロック・インデックスiのデータ・ブロックの受信を続行する。

ステップS3-12の中断時の動作指示待ち状態において受信開始／終了ボタン6が押下されるとステップS3-14へ制御が移り、ここで現在まで受信したデータ・ブロックの登録を行なう。

第25図は第17図で示した受信データ・ブロッ

クが最大登録数20未満であることを判定し（ステップS3-15-1）、条件を満たさない場合はステップS3-6へ移行し、速やかに受信動作を終了する。条件を満足した場合は、候補となる未使用のブロック・インデックスxの初期値として“1”を代入して（ステップS3-15-2）、ブロック・インデックスxのデータ・ブロックがすでに登録済みか否かを判定し（ステップS3-15-3）、未登録であった場合は、受信バッファでのスタート・アドレスBLOCKADR[x]を設定し（ステップS3-15-5）、ブロック・インデックスRBLOCKIにxを格納する（ステップS3-15-6）。一方、S3-15-3の判定において登録済みであった場合は候補となるブロック・インデックスxに1を加えた後（ステップS3-15-4）、再びステップS3-15-3の判定を繰り返す。こうして新たにコンピュータから受信するデータ・ブロックのブロック・インデックスRBLOCKIが設定されたらステップS3-17へ進む。

一方、第17図ステップS3-12の中断時の動作

ク登録S3-14での動作制御のフローチャートである。ここでのフローは前記受信データ・ブロック登録ステップS3-8のものと同様で、第25図に示すように登録済ブロック数RBLOCKSに1を加え（ステップS3-14-1）、受信順テーブル（RBLOCKSEQ[RBLOCKS]）にブロック・インデックスiを（=RBLOCKI）を格納し、（ステップS3-14-2）、点滅中のブロック・インデックスiに対応するブロック・インデックス表示ランプ18を点燈させた後（ステップS3-14-3）、ステップS3-15へ移行する。

第26図は第17図で示した次ブロック・インデックス設定ステップS3-15での動作制御のフローチャートである。

ステップS3-15では次にコンピュータから受信されるデータ・ブロックに与えるブロック・インデックスRBLOCKIの設定を行なう。ここでのフローは前記次ブロック・インデックス設定ステップS3-9のものと同様で、第26図に示すように登録済みのデータ・ブロック数RBLOCKS

指示待ち状態においてクリア・ボタン5が押下されるとS3-16へ制御が移り、ここで現在まで受信したデータ・ブロックの強制消去を行なう。第27図は第17図で示した受信データ・ブロック消去S3-16での動作制御のフローチャートである。

ここでのフローは前記受信データ・ブロック消去ステップS3-10のものと同様で、第27図に示すようにまず残受信バッファ量RREMの値をブロック・インデックスiのデータ・ブロックの受信開始時の状態に戻し（ステップS3-16-1）、復元した残受信バッファ量RREMの受信バッファのサイズUMEMに対する割合を百分率で残バッファ／未送信データ容量表示ランプ17に表示し（ステップS3-16-2）、ステップS3-16-3でバッファ空状態かどうかを判定し、バッファ空状態ならばステップS3-16-4においてバッファ空表示ランプ15を点燈する。そして、現在までステップS3-7で受信していたブロック・インデックスiのデータ・ブロックのサイズを0にクリアし（ステップS3-16-5）、点滅中のブロック・

インデックス*i*に対応するブロック・インデックス表示ランプ18を消燈させた後（ステップS3-16-6）、ステップS3-17へ進む。

第28図は第17図で示した指示待ちステップS3-17での動作制御フローチャートである。

ステップS3-17は受信動作中断処理後における動作指示待ち状態であり、第28図に示すように動作指示のボタンのいずれかが押下されるまで待機している。受信開始／終了ボタン6が押下された場合（ステップS3-17-1）、受信動作の中止とみなし、ステップS3-6へ移行し受信動作は終了する。一方受信中断ボタン7が押下された場合（ステップS3-17-2）、受信動作再開として扱い、ステップS3-18へ移行し、ここで点滅中の受信中表示ランプ13を点燈させユーザーに受信動作が再開したことを知らせた後再びステップS3-3に戻り、新たなデータ・ブロックの最初のバイト・データの受信を行なう。

第17図において前記開始バイト受信ステップS3-3あるいはデータ・ブロック受信ステップ

S3-20での動作制御のフローチャートである。

ステップS3-20はバッファ・フル時における動作指示待ち状態であり、第30図に示すように動作指示ボタンのいずれかが押下されるまで待機している。受信開始／終了ボタン6が押下された場合（ステップS3-20-1）、制御はステップS3-21に移行し、クリア・ボタン5が押下された場合（ステップS3-20-2）、制御はステップS3-23に移行する。

ステップS3-20のバッファ・フル時の動作指示待ち状態において受信開始／終了ボタン6が押下されるとステップS3-21へ制御が移り、ここでバッファ・フル状態になるまで受信していたデータ・ブロックの登録を行なう。

第31図は第17図で示した受信データ・ブロック登録ステップS3-21での動作制御のフローチャートである。

ここでのフローは前記受信データ・ブロック登録ステップS3-8のものと同様で、第31図に示すように登録済ブロック数RBLOCKSに1を

S3-7においてバッファ・フル状態になったと判定された場合には、ステップS3-4に制御が移り、ここで点燈中の受信中表示ランプ13を点滅させユーザーに受信動作が中断中であることを知らせた後ステップS3-19へ進む。

第29図は第17図で示したバッファ・フル状態通知ステップS3-19での動作制御のフローを表わしたものである。

ステップS3-19ではバッファ・フル状態の通知を行なう。第29図に示すようにまずステップS3-19-1でバッファ・フル表示ランプ16を点燈させユーザーにバッファ・フル状態に陥ったことを知らせ、ステップS3-19-2で対コンピュータ制御信号線PE_Cを、ステップS3-19-3で対コンピュータ制御信号線-ERROR_Cをそれぞれアサートすることによりコンピュータに、当該プリンタ・バッファが“紙なし状態”に陥ったことを知らせる。その後ステップS3-20へ移行する。

第30図は第17図で示した指示待ちステップ

加え（ステップS3-21-1）、受信中テーブル(RBLOCKSEQ [RBLOCKS])にブロック・インデックス*i*(=RBLOCKI)を格納し（ステップS3-21-2）、点滅中のブロック・インデックス*i*に対応するブロック・インデックス表示ランプ18を点燈させた後（ステップS3-21-3）、ステップS3-22へ移行する。

第32図は第17図で示した次ブロック・インデックス設定ステップS3-22での動作制御のフローチャートである。

ステップS3-22では次にコンピュータから受信されるデータ・ブロックに与えるブロック・インデックスRBLOCKIの設定を行なう。ここでのフローは前記次ブロック・インデックス設定ステップS3-9のものと同様で、第32図に示すように登録済みのデータ・ブロック数RBLOCKSが最大登録数20未満であることを判定し（ステップS3-22-1）、条件を満たさない場合はステップS3-6へ移行し、速やかに受信動作を終了する。条件を満足した場合は候補となる未使用のブロッ

ク・インデックスxの初期値として“1”を代入して(ステップS3-22-2)、ブロック・インデックスxのデータ・ブロックがすでに登録済みか否かを判定し(ステップS3-22-3)、未登録であった場合は受信バッファでのスタート・アドレスBLOCKADR[x]を設定し(ステップS3-22-5)、ブロック・インデックスRBLOCKIにxを格納する(ステップS3-22-6)。一方ステップS3-22-3の判定において登録済みであった場合は候補となるブロック・インデックスxに1を加えた後(ステップS3-22-4)、再びステップS3-22-3の判定を繰り返す。こうして新たにコンピュータから受信するデータ・ブロックのブロック・インデックスRBLOCKIが設定されたらステップS3-6へ移行し、速やかに受信動作を終了する。

ステップS3-21のバッファ・フル時の動作指示待ち状態においてクリア・ボタン5が押下されるとステップS3-23へ制御が移り、ここでバッファ・フル状態の通知の停止を行なう。

第35図は、第34図で示した送信動作条件判定ステップS4-1での動作制御のフローチャートである。

第35図を見るとわかるようにここでは登録済ブロック数と対プリンタ・ポート制御信号線SLCT_Pの判定が行なわれる。登録済ブロック数の判定ステップS4-1-1では登録済ブロック数RBLOCKSが0を越えているか否か、すなわち

$$RBLOCKS > 0$$

を判定し、さらにSLCT_Pの判定ステップS4-1-2ではSLCT_Pがネゲート状態であることすなわち

$$SLCT_P = \text{"Low"}$$

を判定する。この2つの条件をいずれも満たす場合制御はステップS4-2に移行し、いずれかが満たされなかった場合は送信動作ステップS4を終了する。

ステップS4-2では送信動作の準備を行なう。

第36図は、第34図で示した送信動作準備ステップS4-2での動作制御のフローを表わしたもの

第33図は第17図で示したバッファ・フル状態通知停止ステップS3-23での動作制御のフローチャートである。

第33図に示すようにまずステップS3-23-1で対コンピュータ制御信号線—ERROR_Cを、ステップS3-23-2で対コンピュータ制御線PE_Cをそれぞれネゲートすることによりコンピュータに当該プリンタ・バッファが“紙なし状態”を回避したことを知らせ、さらにステップS3-23-3で点燈中のバッファ・フル表示ランプ16を消燈させユーザーにバッファ・フル状態で無くなったことを知らせた後、バッファ・フル状態になるまで受信していたデータ・ブロックの強制消去を行なうべく、受信データ・ブロック消去ステップS3-16へ移行する。

以上が受信動作ステップS3のフローである。

第34図は、第14図で示した送信動作S4での動作制御の概略を表わしたものである。送信動作ステップS4では、まずステップS4-1で送信動作条件判定を行なう。

である。第36図を見るとわかるように、まずステップS4-2-1で残バッファ/未送信データ容量表示ランプ17を消燈し、ステップS4-2-2で送信中表示ランプ14を点滅させ、ユーザーに送信動作が開始したことを知らせる。次にステップS4-2-3で送信データ・ブロック総数TBLOCKSを0クリアし、ステップS4-2-4で対プリンタ・ポート制御信号線—SLCTIN_Pに前記ステップS1-2において格納したSLINの値を出力し、ステップS4-2-5で対プリンタ・ポート制御信号線—AUTOFEEDXT_Pに前記ステップS1-2において格納したAFDXTの値を出力する。すなわち、ステップS4-2-3～S4-2-5

$$TBLOCKS \leftarrow 0$$

$$-SLCTIN_P \leftarrow SLIN$$

$$-AUTOFEEDXT_P \leftarrow AFDXT$$

という操作を行なう。

そして次にステップS4-2-6で対プリンタ制御信号線—INIT_PをアサートしステップS4-2-7で—INIT_Pをネゲートすることにより“Low”

パルス信号を送出し、対プリンタ・ポートに接続されているプリンタをイニシャライズする。

次に制御はステップS4-3に移行し送信ブロック選択入力待ち状態になる。

第37図は第34図で示した送信ブロック選択入力待ちステップS4-3での動作制御のフローチャートである。

すなわち、送信ブロック選択入力待ちステップS4-3では、第37図に示すように動作指示のボタンのいずれかが押下されるまで待機している。

送信開始/終了ボタン8が押下された場合(ステップS4-3-1)、送信動作の終了とみなし制御はステップS4-16へ移行し、送信動作は終了する。またクリア・ボタン5が押下された場合(ステップS4-3-2)、制御はステップS4-4へ移行し送信順テーブルの消去を行なう。また20個存在するブロック・インデックス・ボタン10のいずれかが押下された場合(ステップS4-3-3)、制御はステップS4-5へ移行し送信選択ブロックの判定を行なう。(便宜的にステップS4-3-3で押下

アする。この後、制御は再びステップS4-3へ戻る。

前記送信ブロック選択入力待ち状態ステップS4-3においてブロック・インデックス・ボタン10が押下されると、制御はステップS4-5に移行し、送信選択されたブロック・インデックスxの判定を行なう。

第39図は、第34図で示した送信選択ブロック判定ステップS4-5での動作制御のフローを表わしたものである。

第39図を見るとわかるように判定条件は、ブロック・インデックスxで参照されるデータ・ブロックが受信登録済であること。すなわち該データ・ブロックのサイズBLOCKSIZE[x]が1(バイト)以上であること

$BLOCKSIZE[x] > 0$ (ステップS4-5-1) 及びブロック・インデックスxで参照されるデータ・ブロックが送信登録済でないこと、すなわち送信順テーブルTBLOCKSEQ[i] ($i=1, 2, \dots, 20$) に該データ・ブロック・インデックスxが登録され

されたブロック・インデックス・ボタン10に対応するブロック・インデックスをxとする) また、全ブロックボタン11が押下された場合(ステップS4-3-4) 制御はステップS4-6へ移行し、全受信済ブロックの登録を行なう。また中断ボタン9が押下された場合(ステップS4-3-5) 制御はステップS4-7へ移行し送信順テーブルの判定を行なう。

前記送信ブロック選択入力待ちステップS4-3においてクリア・ボタン5が押下されると制御はステップS4-4に移行し、送信順テーブルTBLOCKSEQ[i] ($i=1, 2, \dots, 20$) の消去が行なわれる。

第38図は第34図で示した送信順テーブル消去ステップS4-4での動作制御のフローチャートである。

第38図を見るとわかるようにステップS4-4-1からステップS4-4-4において、

$TBLOCKSEQ[i] \leftarrow 0$ ($i=1, 2, \dots, 20$) を実行することにより、送信順テーブルを0にクリ

ていないこと

$TBLOCKSEQ[i] \neq x$ ($i=1, 2, \dots, 20$) (ステップS4-5-2~S4-5-5)

の2つである。

いずれかが満たされなかった場合、制御は再びステップS4-3へ戻り送信ブロック選択入力待ち状態になる。

一方、この2つの条件がいずれも満たされる場合、制御はステップS4-6へ移行し、ブロック・インデックスxは送信順テーブルに登録される。

第40図は第34図で示した送信選択ブロック登録ステップS4-6での動作制御のフローを表わしたものである。

ステップS4-6では送信選択されたブロック・インデックスxの送信順テーブルへの登録が行なわれる。第40図を見るとわかるようにここではまず送信選択されたブロック・インデックスxが送信動作後、何番目に選択されたのかを知り得るためにステップS4-6-1で送信順番iの初期値として“1”を代入する。次にステップS4-6-2で送信

順テーブルの i 番目の領域 TBLOCKSEQ [i] の内容が“0”であるか否かを判定する“0”であった場合、ブロック・インデックス x は i 番目に送信選択されたものであるとしてステップ S4-6-4に進み、“0”でなかった場合、送信順テーブルを次々と検索すべくステップ S4-6-3で送信順番 i に“1”を加えた後、再びステップ S4-6-2の判定を繰り返す。このステップ S4-6-1からステップ S4-6-4での送信順番 i の検索方法において送信順バッファに登録済のブロック・インデックス数は、前記ステップ S4-5の判定により19以下となっているため必ず TBLOCKSEQ [i] = 0 なる i が存在する。

ステップ S4-6-4では、送信選択されたブロック・インデックス x を送信順テーブルの i 番目の領域に登録する。すなわち

TBLOCKSEQ [i] $\leftarrow x$

を実行する。次に制御はステップ S4-6-5に移行し、送信選択されたブロック・インデックス x に対応するブロック・インデックス表示ランプ18を

現在、送信選択されていることを知らせる。すなわちステップ S4-7-1で順番を表わす便宜的なカウンタ“ i ”に初期値として“1”を代入し、続くステップ S4-7-2で受信順テーブルの i 番目の領域 RBLOCKSEQ [i] の内容を読み出し便宜的に x としておく。つまり受信順番 i のデータ・ブロックに対応するブロック・インデックスは x となる。ステップ S4-7-3では、前記ステップ S4-7-2で読み出されたブロック・インデックス x が0で無いことを判定する。もし0でなければステップ S4-7-4に移行し、受信順テーブルの i 番目の領域 RBLOCKSEQ [i] の内容を送信順テーブルの i 番目の領域 TBLOCKSEQ [i] にコピーする。そして次にステップ S4-7-5に進み、点燈中のブロック・インデックス x に対応するブロック・インデックス表示ランプ18を点滅させ、ステップ S4-7-6で受信順番で次のデータ・ブロックを選択すべく順番 i に“1”を加えた後、再びステップ S4-7-2で制御に戻る。一方、ステップ S4-7-3の判定で0であった場合は、受信順テーブル内

点滅させ、ユーザーにブロック・インデックス x のデータ・ブロックが送信選択されたことを知らせる。そしてこの後、制御は再びステップ S4-3へ戻り、送信選択入力待ち状態になる。

前記送信ブロック選択入力待ち状態ステップ S4-3において全ブロック選択ボタン11が押下されると制御は S4-7へ移行し、全受信登録済ブロック・インデックスの送信順テーブルへの登録が行なわれる。

第41図は、第34図で示した全ブロック登録ステップ S4-7での動作制御のフローチャートである。

第41図を見るとわかるようにここでは受信順テーブルの内容を送信順テーブルにコピーすることにより、受信登録済のブロック・インデックスを受信された順番に従い送信順テーブルに登録し、またここで点燈中の現在受信登録されているすべてのデータ・ブロックに対応するブロック・インデックス表示ランプを点滅させ、ユーザーにすべての受信登録されているデータ・ブロックが

の有効データの終了を意味するのでステップ S4-7の動作を終了させ、制御を再びステップ S4-3へ移行し、送信ブロック選択入力待ち状態になる。

前記送信ブロック選択入力待ち状態ステップ S4-3において中断ボタン9が押下されると制御はステップ S4-8へ移行し、送信順テーブルの判定を行う。判定は送信順テーブルの第1番目の領域 TBLOCKSEQ [1] の内容が0であるか否かによって行なう。すなわち

TBLOCKSEQ [1] $\neq 0$

を満足する場合は少なくとも1つ以上のブロック・インデックスが送信順テーブルに登録されているわけであるから送信動作を行うべく制御をステップ S4-9へ移行する。一方、上記条件を満たさなかった場合は、送信順テーブルには現在1つもブロック・インデックスが登録されていない訳であるから制御を再びステップ S4-3に戻し送信ブロック選択入力待ち状態になる。

第42図は、第34図で示した送信動作開始ステップ S4-9での動作制御のフローチャートで

ある。

ステップS4-9では、送信動作が開始させられる。第42図を見るとわかるようにステップS4-9-1でブロック・インデックス表示ランプ18を全消燈し、ステップS4-9-2で送信データ・ブロックサイズ総数TSIZを0でクリアしておく、次にステップS4-9-3では送信順番を表わす便宜的なカウンタ“i”を設け初期値として“1”を代入し、続くステップS4-9-4で送信順テーブルのi番目の領域TBLOCKSEQ[i]の内容を読み出し、便宜的にxとしておく。つまり送信順番iのデータ・ブロックに対応するブロック・インデックスはxとなる。次にステップS4-9-5で前記ステップS4-9-4で読み出されたブロック・インデックスxが0で無いことを判定する。もし0でなければ制御をステップS4-9-6に移行しブロック・インデックスxに対応するブロック・インデックス表示ランプ18を点燈させる。そして次にステップS4-9-7で送信データ・ブロックサイズ総数TSIZにブロック・インデックスxの

に送信すべきデータ・ブロック・インデックスの設定を行なう。すなわち第1番目に送信選択されたデータ・ブロック・インデックスTBLOCKSEQ[1]の内容をTBLOCKIに設定する。次に、ステップS4-9-11で現在点滅中の送信中表示ランプ14を点燈にし、制御をステップS4-10へ移行する。

第43図は、第34図で示したデータ・ブロック送信ステップS4-10での動作制御のフローチャートである。

ステップS4-10では、ブロック・インデックスTBLOCKIのデータ・ブロックの送信を行なう。第43図を見るとわかるように、ここではまずステップS4-10-1で送信データ・ブロック総数TBLOCKSに1を加える。次にステップS4-10-2でブロック・インデックスTBLOCKIに対応するブロック・インデックス表示ランプ18を点燈させ、ユーザーに現在ブロック・インデックスTBLOCKIのデータ・ブロックの送信中であることを知らせ、次にステップS4-10-3で送信中

データ・ブロックサイズBLOCKSIZ[x]を加えて設定し直し、ステップS4-9-8で送信順番で次のデータ・ブロックを選択すべく送信順番iに1を加えた後、再びステップS4-9-4に制御を戻す。

一方ステップS4-9-5の判定で0であった場合は、送信順テーブル内の有効データの終了を意味するので制御をステップS4-9-9へ移行する。こうして現在送信登録されているすべてのデータ・ブロックに対応するブロック・インデックス表示ランプ18を点燈させ、ユーザーに送信選択されているすべてのデータ・ブロックのインデックスを知らせると共に、現在送信登録されているデータ・ブロックのサイズの総数をTSIZに設定する。次にステップS4-9-9では、設定されたTSIZの内容を未送信バイトデータ量TREMにコピーする。これは、現在まだプリンタへ1バイトのデータも送信していないため、TSIZ=TREMとなるためである。次にステップS4-9-10で送信データ・ブロック・インデックスTBLOCKIに第1番目

データ・ブロックの送信済バイト数TBLOCKSIZを0でクリアしておく。ここで送信開始/終了ボタン8が押下された場合(ステップS4-10-4)、送信動作の中止とみなし、制御はステップS4-17へ移行し送信動作は中止する。また中断ボタン9が押下された場合(ステップS4-10-5)、制御はステップS4-15へ移行し、送信中断入力待ち状態となる。いずれのボタンも押下されていない場合、制御はステップS4-10-6に移行し対プリンタ・ポートに接続されたプリンタがエラー状態となっているか否かを対プリンタ制御信号線の状態を見ることにより判定する。判定条件はPE__Pがネゲート状態であること、すなわち

PE__P = "Low" (ステップS4-10-6)

SELECT__Pがアサート状態であること、すなわち

SELECT__P = "High" (ステップS4-10-7)

およびERROR__Pがアサート状態であること、すなわち

ERROR__P = "Low" (ステップS4-10-8)

である。上記条件がすべて満たされた場合、制御はステップS4-10-9へ移行する。一方、いずれかが満たされなかった場合、制御はステップS4-15へ移行し、送信中断入力待ち状態となる。ステップS4-10-9では対プリンタ・ポートに接続されたプリンタが受信可能状態であるか否かを、対プリンタ・ポート制御信号線のBUSY_Pを見ることにより判定する。ここでBUSY_Pがアサート状態である場合、プリンタは受信不可能状態であることを意味するから、制御はステップS4-10-10へ移行し、単位時間ウェイト状態となる。そしてこの後、再び制御は、ステップS4-10-9へ戻り判定を繰り返す。こうしてプリンタが受信可能状態となるのを待つ。一方BUSY_Pがネゲート状態である場合、プリンタは受信可能状態であることを意味するから制御はステップS4-10-11へ移行し、データ送信が行なわれる。

S4-10-11では、受信バッファ内に格納されているバイトデータを対プリンタ・ポート制御信号線DATA0_P~DATA7_Pに送出する。こ

ら1が減じられる。

続くステップS4-10-16では、未送信バイト・データ量のTREMの送信データ・ブロックサイズの総数TSIZに対する割合を百分率で残バッファ/未送信データ容量表示ランプ17に表示する。ここで割合の百分率は次の式によって算出する。

TREM = 0 のとき

未送信データ割合 = 0

TREM ≠ 0 のとき

$$\text{未送信データ割合} = \text{整数部} \left(\frac{\text{TREM}}{\text{TSIZ}} \times 100 \right) + 1$$

次に制御はステップS4-10-17に移行し、現在送信中のデータ・ブロックにおいて未送信データがまだ残っているか否かを判定する。判定は、現在送信中のデータ・ブロックのサイズBLOCKSIZ [TBLOCK I] の内容と送信中データ・ブロック送信済バイト数TBLOCKSIZの内容とを比較することにより行なわれる。すなわち

$$\text{BLOCKSIZ [TBLOCK I]} \neq \text{TBLOCKSIZ}$$

で送信すべきバイト・データのアドレスは、現在送信中のブロック・インデックスTBLOCK Iのデータ・ブロックのスタートアドレスBLOCKADR [TBLOCK I] と送信中データ・ブロックの送信済バイト数TBLOCKSIZの和で得られる。すなわち送信すべきバイト・データの格納されているアドレスは、

$$\text{BLOCKADR [TBLOCK I]} + \text{TBLOCKSIZ}$$

で表わされ、このアドレスに格納されているバイト・データが対プリンタ・ポート制御信号線に送出される。

続いて、この後すぐに対プリンタ制御信号線-STROBE_Pのアサート(ステップS4-10-12) → -STROBE_Pのネゲート(ステップS4-10-13)が行なわれ、プリンタにLowパルス信号が送信される。ここで1バイトのデータの送信が終了したので、次にステップS4-10-14において送信中データ・ブロックの送信済バイト数TBLOCKSIZに1が加えられ、ステップS4-10-15において未送信バイト・データ量TREMか

の場合未送信データがまだ残っていることを意味するから制御は再びステップS4-10-4に戻り次のバイト・データの送信を開始する。一方、

$$\text{BLOCKSIZ [TBLOCK I]} = \text{TBLOCKSIZ}$$

の場合送信中データ・ブロックのすべてのデータが送信されたことを意味するから制御はデータ・ブロックの送信を終了し、ステップS4-11へ移行する。ステップS4-11ではデータ・ブロック送信終了処理が行なわれる。すなわち点滅中のブロック・インデックスTBLOCK Iに対応するブロック・インデックス表示ランプ18を消燈させ、ユーザーにブロック・インデックスTBLOCK Iのデータ・ブロックの送信が終了したことを知らせる。

次に処理は、S4-12に移行し、次送信データ・ブロックの選択が行なわれる。

第44図は第34図で示した次ブロック選択S4-12での動作制御のフローチャートである。

第44図を見るとわかるように、ここではまずステップS4-12-1で送信順テーブルのTBLOCKS

+1番目の領域TBLOCKSEQ(TBLOCKS+1)の内容を読みし便宜的にxとしておく。次にステップS4-12-2で読み出されたブロック・インデックスxが0であるか否かを判定する。もし0でなければ、xが次に送信すべきデータ・ブロックのインデックスということになり、制御をステップS4-12-3へ移行し、送信データ・ブロックのインデックスを表わすTBLOCKIにブロック・インデックスxを格納する。そしてこの後、制御を再びステップS4-10へ戻し次データ・ブロックの送信を開始する。一方ステップS4-12-2の判定でxが0であった場合は、送信順テーブルに登録されたデータ・ブロックはすべて送信が終了して次送信データ・ブロックは存在しないことを意味するため制御をステップS4-13へ移行し送信動作中断入力待ち状態となる。

第45図は、第34図で示した送信動作中断入力待ちS4-13での動作制御のフローチャートである。

ステップS4-13では、第45図を見るとわかる

ステップS4-14-6で残バッファ/未送信データ容量表示ランプ17を消燈する。次にステップS4-14-7からステップS4-14-10において現在受信登録されているすべてのデータ・ブロックに対応するブロック・インデックス表示ランプを点燈させ、ユーザーにすべての受信登録済データ・ブロックのインデックスを知らせる。すなわちステップS4-14-7で受信順番を表わす便宜的なカウンタ“j”に初期値として“1”を代入し、続くステップS4-14-8で受信順テーブルのj番目の領域RBLOCKSEQ[j]の内容を読み出し、便宜的にxとしておく。つまり受信順番jのデータ・ブロックに対応するブロック・インデックスはxとなる。次にステップS4-14-9では、前記ステップS4-14-8で読み出されたブロック・インデックスが0であるか否かを判定する。もし0でなければ制御をステップS4-14-10に移行しブロック・インデックスxに対応するブロック・インデックス表示ランプ18を点燈させる。そして次にステップS4-14-11で受信順番で次のデータ・ブロック

ように、まずステップS4-13-1で現在点燈中の送信中表示ランプ14を点滅させユーザーに送信動作が中断されていることを知らせる。そしてこの後、動作ボタンのいずれかが押下されるまで待機している。ここで送信開始/終了ボタン8が押下された場合(ステップS4-13-2)送信動作の中止終了とみなし、制御はステップS4-17へ移行する。また中断ボタン9が押下された場合(ステップS4-13-3)制御は、ステップS4-14へ移行し再送信の準備が行なわれる。

第46図は、第34図で示した再送信準備ステップS4-14での動作制御のフローを表わしたものである。

第46図を見るとわかるように、ここではまずステップS4-14-1からステップS4-14-4において

TBLOCKSEQ[i] ← 0 (i=1, 2, ..., 20)

を実行することにより送信順テーブルを0にクリアする。次にステップS4-14-5において送信データ・ブロック総数TBLOCKSを0にクリアし、

を選択すべく受信順番jに1を加えた後、再びステップS4-14-8に制御を戻す。一方、ステップS4-14-9の判定でxが0であった場合は受信順テーブル内の有効データの終了を意味するのでステップS4-14の制御を終了し、制御をステップS4-3に戻し再び送信選択入力待ち状態となる。

前記、データ・ブロック送信ステップS4-10において中断ボタン9が押下された場合、あるいは同じく、ステップS4-10において対プリンタポートに接続されたプリンタのエラー状態が検出された場合、制御はステップS4-15へ移行し送信中断入力待ち状態になる。

第47図は第34図で示した、送信中断入力待ちS4-15での動作制御のフローを表わしたものである。

第47図を見るとわかるようにここでは、まずステップS4-15-1で現在点燈中の送信中表示ランプ14を点滅させユーザーに送信が中断されていることを知らせる。そしてこの後動作ボタンのいずれかが押下されるまで待機している。ここで

送信開始／終了ボタン8が押下された場合（ステップS4-15-2）送信動作の中止終了とみなし制御はステップS4-17へ移行する。またクリア・ボタン5が押下された場合（ステップS4-15-3）制御はステップS4-16へ移行し送信中断中のデータ・ブロック送信をスキップする。また、中断ボタン9が押下された場合（ステップS4-15-4）、制御はステップS4-15-5に移行し点滅中の送信中表示ランプ14を点燈させ、送信中断されていたデータ・ブロックの送信を再開すべく制御を再びステップS4-10へ戻す。

前記送信中断入力待ち状態ステップS4-15においてクリア・ボタン5が押下されると制御はステップS4-16へ移行し送信中断中データ・ブロック送信のスキップを行なう。この処理は送信中断されたデータ・ブロックにおいて、何バイトのデータが未送信であったかを計算し、その値を未送信データバイト量TREMから減じることにより行なわれる。

送信中断データ・ブロックにおける未送信デー

S4-17での動作制御のフローチャートである。

第48図を見るとわかるように、ここではまずS4-17-1からS4-17-4において

$$TBLOCKSEQ[i] \leftarrow 0 \quad (i=1, 2, \dots, 20)$$

を実行することにより送信順テーブルを0にクリアする。次にステップS4-17-5において残受信バッファ量RREMの受信バッファのサイズUMEMに対する割合を百分率で残バッファ／未送信データ容量表示ランプ17に表示する。割合の百分率の算出式は前記受信動作ステップS3-3-13で用いたものと同じである。次にステップS4-17-6からステップS4-17-10において現在受信登録されているすべてのデータ・ブロックに対応する、ブロック・インデックス表示ランプを点燈させユーザーにすべての受信登録済データ・ブロックのインデックスを知らせる。この処理の方法は前記再送信準備ステップS4-14のステップS4-14-7からステップS4-14-11における処理とまったく同様である。この後処理はステップS4-18に移行する。ステップS4-18では送信中表示

タ量を求めるには、送信中断されたデータ・ブロックのサイズBLOCKSIZ [TBLOCK I]から送信中データ・ブロックの送信済バイト数TBLOCKSIZを減ずればよい。すなわち

$$BLOCKSIZ [TBLOCK I] - TBLOCKSIZ$$

を計算すればよい。

よってここでは

$$TREM \leftarrow TREM - (BLOCKSIZ [TBLOCK I] - TBLOCKSIZ)$$

を実行する。そしてこの後制御をS4-10へ移行し、送信中断データ・ブロックの送信を再開させる。

前記データ・ブロック送信ステップS4-10において、送信開始／終了ボタン8が押下された場合、前記送信動作中断入力待ち状態S4-13において送信開始／終了ボタン8が押下された場合、あるいは前記送信中断入力待ち状態S4-15で送信開始／終了ボタン8が押下された場合には、制御はステップS4-17へ移行し送信中止となる。

第48図は、第34図で示した送信中止ステップ

ランプ14を消燈しすべての送信動作を終了する。

第49図は第14図で示したデータ・ブロックの選択的クリア動作ステップS5での動作制御の概略を表わしたものである。選択的クリア動作ステップS5は、第49図に示すようなフローで行なわれる。すなわち、まずステップS5-1で登録済ブロック数の判定を行なう。以下に述べるデータ・ブロックの選択的クリア動作ステップS5は現在コンピュータから受信登録されているデータ・ブロックを対象にしている訳であるから、少なくとも登録済ブロック数が1以上であることが必要になる。すなわちこの登録済ブロック数判定ステップS5-1では登録済ブロック数RBLOCKSに対して

$$RBLOCKS > 0$$

を満足するか否かを判定し、満足する場合はステップS5-2に移行し、条件を満たさなかった場合はデータ・ブロックの選択的クリア動作ステップS5を終了する。

第50図は第49図で示したブロック・インデックス入力待ちステップS5-2での動作制御のフロー

チャートである。

ステップS5-2は消去するブロック・インデックスの入力待ち状態であり、第50図に示すように動作指示のボタンのいずれかが押下されるまで待機している。クリア・ボタン5が押下された場合は(ステップS5-2-1)、データ・ブロックの選択的クリア動作の終了指示とみなし速やかに当該動作ステップS5を終了する。また、20個存在するブロック・インデックス・ボタン10のいずれかが押下された場合(ステップS5-2-2)、制御はステップS5-3に移行し(便宜的にステップS5-2-2で押下されたブロック・インデックス・ボタン10に対応するブロック・インデックスをxとする)、全ブロック選択ボタン11が押下された場合(ステップS5-2-3)、制御はステップS5-4に移行する。

ステップS5-3では、前記ブロック・インデックス入力待ちステップS5-2でブロック・インデックスボタン10によって入力されたブロック・インデックスxで参照されるデータ・ブロックが

ステップS5-6は消去動作確認待ち状態であり、第51図に示すようにブロック・インデックスxに対応するブロック・インデックス・ボタン10またはクリア・ボタン5のいずれかが押下されるまで待機している。ブロック・インデックスxのブロック・インデックス・ボタン10が押下された場合は(ステップS5-6-1)、ブロック・インデックスxのデータ・ブロックに対する消去動作のキャンセル指示とみなしステップS5-7に移行し、クリア・ボタン5が押下された場合は(ステップS5-6-2)、ブロック・インデックスxのデータ・ブロックに対する消去動作の実行指示とみなしステップS5-8へ移行する。

ステップS5-7では点滅中のブロック・インデックスxに対応するブロック・インデックス表示ランプ18を点燈させ、ユーザーにブロック・インデックスxのデータ・ブロックが消去の対象にされていないことを知らせた後、再びブロック・インデックス入力待ちステップS5-2に制御を戻す。

消去動作確認待ち状態ステップS5-6において、

登録されているか否かを判定する。判定は該データ・ブロックのサイズBLOCKSIZE[x]が1(バイト)以上であるか否かによって行なう。すなわち

$$\text{BLOCKSIZE}[x] > 0$$

を満足する場合は、ブロック・インデックスxのデータ・ブロックは現在登録されている訳であるから消去動作を行なうべくステップS5-5へ進む。一方条件を満たさなかった場合はブロック・インデックスxのデータ・ブロックは現在登録されていない訳であるから再びブロック・インデックス入力待ちステップS5-2に制御を戻す。

ステップS5-5では点燈中のブロック・インデックスxに対応するブロック・インデックス表示ランプ18を点滅させ、ユーザーにブロック・インデックスxのデータ・ブロックが現在消去の対象として選択されていることを知らせた後、ステップS5-6へ進む。

第51図は第49図で示した消去動作確認待ちステップS5-6での動作制御のフローチャートである。

クリア・ボタン5が押下された場合にはステップS5-8に制御が移り、ここでブロック・インデックスxのデータ・ブロックに対する消去が行なわれる。

第52図は第49図で示したデータ・ブロック消去ステップS5-8での動作制御のフローチャートである。

ここでは第52図に示すようにまず消去の対象として選択されたブロック・インデックスxのデータ・ブロックが受信動作開始後何番目(m番目)に受信したデータ・ブロックなのかを知り得るために、ステップS5-8-1で求める受信順番mの初期値として“1”を代入する。次にステップS5-8-2で受信順テーブルのm番目の領域RBLOCKSEQ[m]の内容と、消去の対象として選択されたブロック・インデックスxが一致するか否かを判定し、一致した場合はmがブロック・インデックスxのデータ・ブロックの受信順番であるとしてステップS5-8-4に進み、一致しなかった場合は受信順テーブルを次々と検索すべくステップS5-8-3で受信順番mに1を加えた後再びステップS5-8

—2の判定を繰り返す。このステップS5-8-1からステップS5-8-3での受信順番mの検索方法において消去の対象として選択されたブロック・インデックスxのデータ・ブロックは前記データ・ブロック登録判定ステップS5-3によって現在登録されていることがわかっているので必ずブロック・インデックスxのデータ・ブロックに対する受信順番mが得られる。続くステップS5-8-4では以降ステップS5-8-13に渡って行なわれるところの消去の対象として選択されたブロック・インデックスxのデータ・ブロックよりも受信順番で後に受信されたいくつかのデータ・ブロックに対して受信バッファ内での格納位置をそれぞれずらして消去されるブロック・インデックスxのデータ・ブロックが格納されていた領域に詰める操作の準備として、ブロック・インデックスxのデータ・ブロックよりも後に受信されたいくつかのデータ・ブロックに対する受信順番を表わす便宜的なカウンタ“n”を設け、その初期値としてm+1を代入する。ここで受信順番nの取り得る

ので速やかにステップS5-8-14へ移行することになる。

ステップS5-8-6では受信順番nのデータ・ブロックに与えられたブロック・インデックス“y”を得るために、受信順テーブルのn番目の領域RBLOCKSEQ[n]の内容を読み出す。続くステップS5-8-7ではこの受信順番nのブロック・インデックスyのデータ・ブロックの格納されたそれぞれのバイト・データを計数する便宜的なカウンタ“i”を設け、その初期値として0を代入する。こうしてステップS5-8-8でブロック・インデックスyのデータ・ブロックを構成する各々のバイト・データを、消去されるブロック・インデックスxのデータ・ブロックのサイズであるBLOCKSIZ[x]（バイト）分受信バッファ内で前方（アドレス下位方向）に転送する操作を行う。ここで転送するバイト・データが格納されている領域のアドレスはブロック・インデックスyのデータ・ブロックのスタートアドレスBLOCKADR[y]とカウンタiの和で得られ、新しい格納領域

最大値は現在受信バッファに登録済のデータ・ブロックの総数RBLOCKSであるからこの受信順番nの取り得る値の範囲は

$$m+1 \leq n \leq \text{RBLOCKS}$$

と考えられるが、消去されるブロック・インデックスxのデータ・ブロックが受信順番で最後のデータ・ブロックであった場合、すなわち

$$m = \text{RBLOCKS}$$

であった場合はブロック・インデックスxのデータ・ブロックよりも受信順番で後に受信されたデータ・ブロックは存在しないので以下に述べる格納位置の修正の操作は必要ない。ステップS5-8-5ではデータ・ブロックの格納位置の修正が必要なのか否かを受信順番nが登録済ブロック数RBLOCKS以下であるか否かで判定する。条件を満たす場合はステップS5-8-6に進み、満たさない場合はステップS5-8-14へ進む。前述した消去されるブロック・インデックスxのデータ・ブロックが受信順番で最後のデータ・ブロックであった場合はステップS5-8-5の判定条件を満たさない

（転送先）アドレスは前述の転送元アドレスから消去されるブロック・インデックスxのデータ・ブロックのサイズBLOCKSIZで[x]を引いたものである。すなわち転送元アドレスは、

$$\text{BLOCKADR}[y] + i$$

で表わされ、転送先アドレスは

$$\text{BLOCKADR}[y] - \text{BLOCKSIZ}[x] + i$$

で表わされる。ステップS5-8-9では次のバイト・データを指し示すべくカウンタiに1を加え、ステップS5-8-10でブロック・インデックスyのデータ・ブロックを構成するすべてのバイト・データが転送されたか否かを判定すべくカウンタiがブロック・インデックスyのデータ・ブロックのサイズBLOCKSIZ[y]未満であるかを判定する。条件を満たす場合はまた転送すべきバイト・データが残っている訳であるからバイト・データの転送を行なうべく再びステップS5-8-8に戻り、条件を満たさない場合はすべてのバイト・データの転送が終了した訳であるからステップS5-8-11に進む。ステップS5-8-11では

転送が終了したブロック・インデックス y のデータ・ブロックのスタートアドレスBLOCKADR〔 y 〕から消去されるブロック・インデックス x のデータ・ブロックのサイズBLOCKSIZ〔 x 〕を引いて設定し直す。続くステップS5-8-12で受信順テーブルにおいて転送されたブロック・インデックス y のデータ・ブロックの受信順番をひとつ前にずらす操作を行なう。すなわち

$$RBLOCKSEQ〔n-1〕 \leftarrow RBLOCKSEQ〔n〕$$

を行なう。以上述べてきたステップS5-8-6からステップS5-8-12にかけての操作によって消去されるブロック・インデックス x のデータ・ブロックよりも受信順番で後に受信されたひとつのデータ・ブロックであるところの受信順番 n のデータ・ブロックの格納位置の修正が行なわれた訳であるがステップS5-8-13では受信順番でさらに次のデータ・ブロックの格納位置の修正を行なうべく受信順番 n に1を加えた後再び前記ステップS5-8-5に戻る。受信順番で最後のデータ・ブロックまでのすべてのデータ・ブロッ

クに対する格納位置の修正が終了するとステップS5-8-5での判定条件を満たさなくなるので結果的にステップS5-8-14へ移行する。消去されるブロック・インデックス x のデータ・ブロックが受信順番で最後のデータ・ブロックであった場合、または消去されるブロック・インデックス x のデータ・ブロックよりも受信順番で後に受信されたすべてのデータ・ブロックに対して前述したような格納位置の修正が行なわれた後制御はステップS5-8-14へ移り、ここで結果的に意味の無くなった（空き領域になった）受信順番で最後に受信したデータ・ブロックに対するブロック・インデックスが格納されている受信順テーブルの領域RBLOCKSEQ〔RBLOCKS〕を0でクリアする。次にブロック・インデックス x のデータ・ブロックを消去することによってバッファ・フル状態が回避されるのか否かを判定するためにステップS5-8-15で現在の残受信バッファ量RREMが0以上であるか否かを判定する。条件を満たす場合はブロック・インデックス x のデータ・ブロックを

消去する前もバッファ・フル状態で無い訳であるからステップS5-8-19に進む。条件を満たさない場合はブロック・インデックス x のデータ・ブロックの消去によってバッファ・フル状態から回避される訳であるから、ステップS5-8-16へ進みここで対コンピュータ制御信号線-ERROR_Cを、ステップS5-8-17で、対コンピュータ制御信号線PE_Cをそれぞれネゲートすることによりコンピュータに当該プリンタ・バッファが“紙なし状態”を回避したことを知らせ、さらに、ステップS5-8-18で点燈中のバッファ・フル表示ランプ16を消燈させ、ユーザーにバッファ・フル状態で無くなったことを知らせた後ステップS5-8-19へ進む。ステップS5-8-19では残受信バッファ量RREMに消去されるブロック・インデックス x のデータ・ブロックのサイズBLOCKSIZ〔 x 〕を加えて設定し直す。続くステップS5-8-20ではステップS5-8-19で修正された残受信バッファ量RREMの受信バッファのサイズUMEMに対する割合を百分率で残バッファ／未送信データ容量

表示ランプ17に表示する。ステップS5-8-21ではステップS5-8-19での残受信バッファ量RREMの修正によってバッファ空状態になったか否かを残受信バッファ量RREMの値が受信バッファのサイズUMEMの値と等しいか否かによって判定し、条件を満足する場合すなわちバッファ空状態になった場合はステップS5-8-22でバッファ空表示ランプ15を点燈させステップS5-8-23に進む。一方条件を満足しない場合は速やかにステップS5-8-23へ移行する。ステップS5-8-23ではいよいよ消去の対象として選択されたブロック・インデックス x のデータ・ブロックを消去すべくそのサイズBLOCKSIZ〔 x 〕を0にクリアする。続くステップS5-8-24で登録済ブロック数RBLOCKSから1を減じることによってこのデータ・ブロック消去ステップS5-8は終了しステップS5-9へ移行する。

ステップS5-8におけるブロック・インデックス x のデータ・ブロックに対する消去が終了するとステップS5-9へ制御が移り、ここで点滅中の

ブロック・インデックスxに対応するブロック・インデックス表示ランプ18を消燈させ、ユーザーにブロック・インデックスxのデータ・ブロックが消去されたことを知らせた後ステップS5-10へ進む。

第53図は第49図で示した次ブロック・インデックス設定ステップS5-10での動作制御でのフローチャートである。

ステップS5-10では今後再び受信動作ステップS3が行なわれる場合に備えて次にコンピュータから受信されるデータ・ブロックに与えるブロック・インデックスRBLOCKIの設定を行なう。第53図に示すようにまずステップS5-10-1において現在登録されているブロック数RBLOCKSが0であるか否かを判定し条件を満たす場合は、ステップS5-10-2に移行し、条件を満たさない場合はステップS5-10-4に進む。現在登録されているデータ・ブロックがひとつも存在しない場合は前記イニシャライズ動作ステップS1で設定した様に最初に受信登録されるデータ・ブロック

一方0で無かった場合はブロック・インデックスxのデータ・ブロックは現在登録されている訳であるからステップS5-10-6へ進み、候補となるブロック・インデックスxに1を加えた後再びステップS5-10-5の判定を繰り返す。この方法において使用されていないブロック・インデックスは前記データ・ブロック消去ステップS5-8によって少なくともひとつは存在するので必ず未登録のデータ・ブロックに対するブロック・インデックスxが得られる。このようにして未使用のブロック・インデックスxが見つかりステップS5-10-7で現在受信バッファに登録されているいくつかのデータ・ブロックのなかで、受信順番で最後のデータ・ブロックに対応するブロック・インデックスyを得るために、受信順テーブルのRBLOCKS番目の領域RBLOCKSEQ[RBLOCKS]の内容を読み出す。続くステップS5-10-8で未登録のブロック・インデックスxのデータ・ブロックのスタート・アドレスBLOCKADR[x]に最後に受信登録されたブロック・インデックスyの

に与えるブロック・インデックスにデフォルト値として“1”を与えるべくステップS5-10-2でブロック・インデックス“1”のデータ・ブロックのスタート・アドレスBLOCKADR[1]に受信バッファのスタート・アドレスMEMSTAを設定し、ステップS5-10-3で次に受信されるデータ・ブロックに与えるブロック・インデックスRBLOCKIに1を格納する。一方ステップS5-10-1の判定で現在登録されているデータ・ブロックがいくつか存在する場合は前記受信動作ステップS3での次ブロック・インデックス設定ステップS3-9のフローに従って制御が進められる。まずステップS5-10-4で候補となる未使用のブロック・インデックスXの初期値として“1”を代入する。その後ステップS5-10-5においてブロック・インデックスxのデータ・ブロックが現在登録済みか否かをそのサイズであるところのBLOCKSIZ[x]が0かどうかによって判定する。0の場合はブロック・インデックスxのデータ・ブロックは未登録であるとみなしステップS5-10-7へ進む。

データ・ブロックのスタート・アドレスBLOCKADR[y]とそのサイズBLOCKSIZ[y]の和を設定する。すなわち、

$$\text{BLOCKADR}[x] \leftarrow \text{BLOCKADR}[y] + \text{BLOCKSIZ}[y]$$

の操作を行なう。この後ステップS5-10-9で次に受信されるデータ・ブロックに与えるブロック・インデックスRBLOCKIにXを格納する。ステップS5-10-3あるいはステップS5-10-9のいずれかにおいて新たにコンピュータから受信するデータ・ブロックのブロック・インデックスRBLOCKIが設定されたら再びステップS5-1に戻り、登録済みのデータ・ブロックの総数が0でない限りデータ・ブロックの消去動作を繰り返す。

前記ブロック・インデックス入力待ちステップS5-2で全ブロック選択ボタン11が押下された場合、ステップS5-4へ制御が移り、ここで点燈中の現在登録されているすべてのデータ・ブロックに対応するブロック・インデックス表示ランプ18を点滅させ、ユーザーにすべての登録されてい

るデータ・ブロック現在消去の対象として選択されていることを知らせる。

第54図は第49図で示した全登録ブロック・インデックス表示18点滅ステップS5-4での動作制御のフローチャートである。

フローは第54図に示すように受信順番を表わす便宜的なカウンタ“m”を設けステップS5-4-1で初期値として“1”を代入し、続くステップS5-4-2で受信順テーブルのm番目の領域RBLOCKSEQ〔m〕の内容を読み出し、便宜的にXとしておく。つまり受信順番mのデータ・ブロックに対応するブロック・インデックスはXとなる。ステップS5-4-3では前記ステップS5-4-2で読み出されたブロック・インデックスxが0で無いことを判定する。もし0でなければステップS5-4-4に移行し点燈中のブロック・インデックスxに対応するブロック・インデックス表示ランプ18を点滅させ、ステップS5-4-5で受信順番で次のデータ・ブロックを選択すべく受信順番mに1を加えた後再びステップS5-4-2に制御を戻す。一方、ス

ンデックス表示18点燈ステップS5-12での動作制御のフローチャートである。

ステップS5-12では点滅中の現在登録されているすべてのデータ・ブロックに対応するブロック・インデックス表示ランプ18を点燈させ、ユーザーにいかなるデータ・ブロックも消去の対象にされていないことを知らせる。ここでのフローは前記全登録ブロック・インデックス表示点滅ステップS5-4のものと基本的に同様で、第56図に示すように受信順番mに初期値“1”を代入し（ステップS5-12-1）、受信順番mのデータ・ブロックに対応するブロック・インデックスxを読み出し（S5-12-2）、そのブロック・インデックスxが0で無いことを確認し（ステップS5-12-3）、0で無ければ点滅中のブロック・インデックスxに対応するブロック・インデックス表示ランプ18を点燈させ（ステップS5-12-4）、受信順番mに1を加えた後（ステップS5-12-5）、再びステップS5-12-2に戻る。一方ステップS5-12-3の判定で0であった場合はステップS5-

テップS5-4-3の判定で0であった場合は受信順テーブル内の有効データの終了を意味するのでステップS5-4での動作を終了する。こうして現在登録されているすべてのデータ・ブロックに対するブロック・インデックス表示ランプ18を点滅させたら次にステップS5-11に進む。

第55図は第49図で示した消去動作確認待ちステップS5-11での動作制御のフローチャートである。

ステップS5-11は消去動作確認待ち状態である。第55図に示すように全ブロック選択ボタン11又はクリア・ボタン5のいずれかが押下されるまで待機している。全ブロック選択ボタン11が押下された場合は（ステップS5-11-1）、すべてのデータ・ブロックに対する消去動作のキャンセル指示とみなしステップS5-12に移行し、クリア・ボタン5が押下された場合は（ステップS5-11-2）、すべてのデータ・ブロックに対する消去動作の実行指示とみなしステップS5-13へ移行する。

第56図は第49図で示した全登録ブロック・イ

ンデックス表示18点燈ステップS5-12での動作を終了する。こうして現在登録されているすべてのデータ・ブロックに対するブロック・インデックス表示ランプ18が再び点燈されたら再度ブロック・インデックス入力待ちステップS5-2に制御を戻す。

消去動作確認待ち状態ステップS5-11においてクリア・ボタン5が押下された場合にはS5-13に制御が移り、ここで現在登録されているすべてのデータ・ブロックに対する消去が行なわれる。

第57図は第49図で示した全データ・ブロック消去S5-13での動作制御のフローチャートである。ここでは第57図に示すように、まずステップS5-13-1ですべての受信順テーブルの領域RBLOCKSEQ〔1〕～RBLOCKSEQ〔20〕を0でクリアする。次にこの消去動作によってバッファ・フル状態が回避されるのか否かを判定するためにステップS5-13-2で現在の残受信バッファ量RREMが0以上であるか否かを判定する。条件を満たす場合はこの消去動作の前もバッファ・フル状態で無い訳であるからステップS5-13-6

に進む。条件を満たさない場合はこの消去動作によってバッファ・フル状態から回避される訳であるから、ステップS5-13-3へ進みここで対コンピュータ制御信号線—ERROR—CをステップS5-13-4で対コンピュータ制御信号線PE—Cをそれぞれネゲートすることによりコンピュータに当該プリンタ・バッファが“紙なし状態”を回避したことを知らせ、さらにステップS5-13-5で点燈中のバッファ・フル表示ランプ16を消燈させ、ユーザーにバッファ・フル状態で無くなったことを知らせた後ステップS5-13-6へ進む。ステップS5-13-6では残受信バッファ量RREMをイニシャル時同様にバッファ空状態すなわち受信バッファのサイズUMEMの内容を代入する。続くステップS5-13-7では残受信バッファ量RREMの受信バッファのサイズUMEMに対する割合として“100%”を残バッファ/未送信データ容量表示ランプ17に表示する。ステップS5-13-8でバッファ空表示ランプ15を点燈させた後ステップS5-13-9ですべてのデータ・

バッファのスタート・アドレスMEMSTAが設定される（ステップS5-10-2）。

その後ステップS5-1に移行して登録ブロック数の判定が行なわれるがこの場合現在登録されているデータ・ブロックはひとつも存在しないのでデータ・ブロックの選択的クリア動作ステップS5を速やかに終了する。

以上がデータ・ブロックの選択的クリア動作ステップS5のフローである。

以上が本発明の実施例であるところのプリンタ・バッファの動作制御のフローである。

（以下余白）

ブロックのサイズBLOCKSIZ〔1〕～BLOCKSIZ〔20〕を0でクリアする。続くステップS5-13-10で登録済ブロック数RBLOCKSを0でクリアしてこの全データ・ブロック消去ステップS5-13は終了しステップS5-14へ移行する。

ステップS5-14ではすべてのブロック・インデックス表示ランプ18を消燈させ、ユーザーにすべてのデータ・ブロックが消去されたことを知らせた後、前記次ブロック・インデックス設定ステップS5-10へ進む。

ステップS5-10では前述したように今後再び受信動作ステップS3が行なわれる場合に備えて次にコンピュータから受信されるデータ・ブロックに与えるブロック・インデックスRBLOCKIの設定を行なう。ただしこの場合現在登録されているデータ・ブロックはひとつも存在しないので第53図に示すようにRBLOCKIにデフォルト値“1”が設定され（ステップS5-10-3）またブロック・インデックス“1”のデータ・ブロックのスタート・アドレスBLOCKADR〔1〕に受信

（Ⅲ）以上説明してきた本発明の実施例であるところのプリンタ・バッファにおいて、受信中断ボタン7、受信中表示ランプ13およびデータ・ブロック受信ステップS3-7、受信中表示ランプ13点滅ステップS3-11、指示待ちステップS-12、受信中表示ランプ13点燈ステップS3-13を設けることによってユーザーはコンピュータが設置されている場所でプリンタ・バッファ装置のプリント出力データの受信動作の中断／再開指示を容易に行うことが可能になる他、現在プリンタ・バッファ装置の受信動作が中断状態であることを容易に確認することが可能になる。

また指示待ちステップS3-12、受信データ・ブロック消去ステップS3-16、受信データ・ブロック登録ステップS3-14、次ブロック・インデックス設定ステップS3-15および受信中断ボタン13、クリアボタン5、受信開始／終了ボタン6を設けることによって主メモリ33に蓄積されたすべてのプリント出力データを消去する

ことなく不要となる現在受信中のプリント出力データを含むデータ・ブロックのみを消去することが可能になる。

〔発明の効果〕

以上説明したように、本発明によれば、データ・バッファ装置にデータ送出装置から送信されるデータの受信動作を中断する受信中断手段、現在受信動作が中断状態であることを表示する手段、さらに受信動作を再開する受信再開手段を設けることによって、ユーザはデータの送出装置が設置されている場所でデータ・バッファ装置のデータの受信動作の中断／再開指示を容易に行うことが可能になる他、現在データ・バッファ装置の受信動作が中断状態であることを容易に確認することが可能になる。

また、受信動作の中断状態において現在受信中であったデータを含むブロックを蓄積手段から消去する消去手段、受信動作の中断状態において現在受信中であったデータを含むブロックをひとつのブロックとして登録可能な登録手段及び受信動

作の中断状態において受信動作を再開する受信再開手段を設け、さらに前記受信再開手段、消去手段あるいは前記登録手段のいずれかの動作を選択する手段を設けることによって、蓄積手段に蓄積されたすべてのデータを消去することなく不要となる現在受信中のデータを含むブロックのみを消去することが可能になる。

(以下余白)

4. 図面の簡単な説明

第1図は本発明の実施例のプリンタ・バッファの外観図、

第2図は本発明の実施例のプリンタ・バッファの平面図、

第3図は本発明の実施例のプリンタ・バッファのブロック図、

第4図は本発明の実施例のプリンタ・バッファの使用形態を表わした図、

第5図は本発明の実施例のプリンタ・バッファの各ポートにおけるデータおよび制御信号線の取扱いを表わした図、

第6図は本発明の実施例のプリンタ・バッファにおけるデータ・ブロックの概念図、

第7図は本発明の実施例のプリンタ・バッファにおける主メモリ33の内容を表わした図、

第8図は制御パラメータ・テーブルの構成の概要図、

第9図は第8図で示した環境テーブルの構成図、

第10図は対コンピュータ・ポートの制御信号線

のイニシャル状態STATEのビット構成図、

第11図は第8図で示したブロック登録テーブルの構成図、

第12図は第8図で示した受信順テーブルの内容を表わした図、

第13図は第8図で示した送信順テーブルの内容を表わした図、

第14図は本発明の実施例のプリンタ・バッファにおける動作制御の概略図、

第15図は第14図で示したイニシャライズ動作ステップS1での動作制御のフローチャート、

第16図は第14図で示したアイドル状態ステップS2での動作制御のフローチャート、

第17図は第14図で示した受信動作ステップS3での動作制御の概略図、

第18図は第17図で示した受信動作条件判定ステップS3-1での動作制御のフローチャート、

第19図は第17図で示した開始バイト受信ステップS3-3での動作制御のフローチャート、

第20図は第17図で示したデータ・ブロック

受信ステップS3-7での動作制御のフローチャート、

第21図は第17図で示した受信データ・ブロック登録ステップS3-8での動作制御のフローチャート、

第22図は第17図で示した次ブロック・インデックス設定ステップS3-9での動作制御のフローチャート、

第23図は第17図で示した受信データ・ブロック消去ステップS3-10での動作制御のフローチャート、

第24図は第17図で示した指示待ちステップS3-12での動作制御のフローチャート、

第25図は第17図で示した受信データ・ブロック登録ステップS3-14での動作制御のフローチャート、

第26図は第17図で示した次ブロック・インデックス設定ステップS3-15での動作制御のフローチャート、

第27図は第17図で示した受信データ・ブロッ

ク消去ステップS3-16での動作制御のフローチャート、

第28図は第17図で示した指示待ちステップS3-17での動作制御のフローチャート、

第29図は第17図で示したバッファ・フル状態通知ステップS3-19での動作制御のフローチャート、

第30図は第17図で示した指示待ちステップS3-20での動作制御のフローチャート、

第31図は第17図で示した受信データ・ブロック登録ステップS3-21での動作制御のフローチャート、

第32図は第17図で示した次ブロック・インデックス設定ステップS3-22での動作制御のフローチャート、

第33図は第17図で示したバッファ・フル状態通知停止ステップS3-23での動作制御のフローチャート、

第34図は第14図で示した送信動作^{ステップ}S4での動作制御の概略図、

第35図は第34図で示した送信動作条件判定ステップS4-1での動作制御のフローチャート、

第36図は第34図で示した送信動作準備ステップS4-2での動作制御のフローチャート、

第37図は第34図で示した送信ブロック選択入力待ちステップS4-3での動作制御のフローチャート、

第38図は第34図で示した送信順テーブル消去ステップS4-4での動作制御のフローチャート、

第39図は第34図で示した送信選択ブロック判定ステップS4-5での動作制御のフローチャート、

第40図は第34図で示した送信選択ブロック登録ステップS4-6での動作制御のフローチャート、

第41図は第34図で示した全ブロック登録ステップS4-7での動作制御のフローチャート、

第42図は第34図で示した送信動作開始ステップS4-9での動作制御のフローチャート、

第43図は第34図で示したデータ・ブロック

送信ステップS4-10での動作制御のフローチャート、

第44図は第34図で示した次ブロック選択ステップS4-12での動作制御のフローチャート、

第45図は第34図で示した送信動作中断入力待ちステップS4-13での動作制御のフローチャート、

第46図は第34図で示した再送信準備ステップS4-14での動作制御のフローチャート、

第47図は第34図で示した送信中断入力待ちステップS4-15での動作制御のフローチャート、

第48図は第34図で示した送信中止ステップS4-17での動作制御のフローチャート、

第49図は第14図で示したデータ・ブロックの選択的クリア動作ステップS5での動作制御の概略図、

第50図は第49図で示したブロック・インデックス入力待ちステップS5-2での動作制御のフローチャート、

第51図は第49図で示した消去動作確認待ち

ステップ S5-6 での動作制御のフローチャート、

第 52 図は第 49 図で示したデータ・ブロック
消去ステップ S5-8 での動作制御のフローチャート、

第 53 図は第 49 図で示した次ブロック・イン
デックス設定ステップ S5-10 での動作制御のフ
ローチャート、

第 54 図は第 49 図で示した全登録ブロック・イ
ンデックス表示 18 点減ステップ S5-4 での動作制
御のフローチャート、

第 55 図は第 49 図で示した消去動作確認待ち
ステップ S5-11 での動作制御のフローチャート、

第 56 図は第 49 図で示した全登録ブロック・イ
ンデックス表示 18 点増ステップ S5-12 での動作
制御の フローチャート、

第 57 図は第 49 図で示した全データ・ブロック
消去ステップ S5-13 での動作制御のフローチャー
ト、

第 58 図は一般的なコンピュータとプリンタ間の
データ送受のための信号線を表わした図、

第 59 図は一般的なコンピュータとプリンタ間
のデータ送受のための主要信号線のタイミング
チャート、

第 60 図は一般的なコンピュータとプリンタ間の
データ送受のためのコンピュータ側におけるプリ
ント出力データの送信動作のフローチャート、

第 61 図は一般的なコンピュータとプリンタ間の
データ送受のためのプリンタ側におけるプリント
出力データの受信動作のフローチャート、

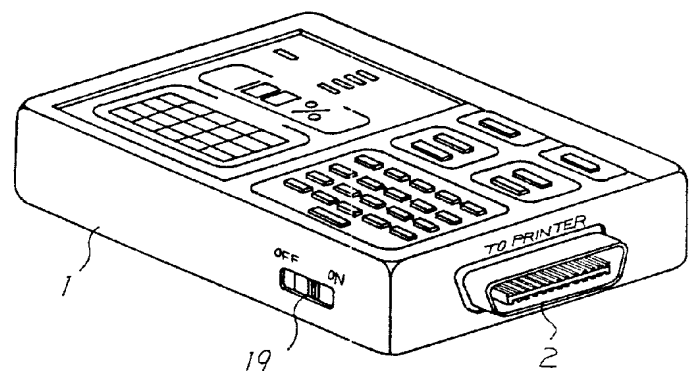
第 62 図は従来のプリンタ・バッファ装置の使用
形態を表わした図、

第 63 図は従来のプリンタ・バッファ装置におけ
るプリント出力データの送受信動作のフローチャー
トである。

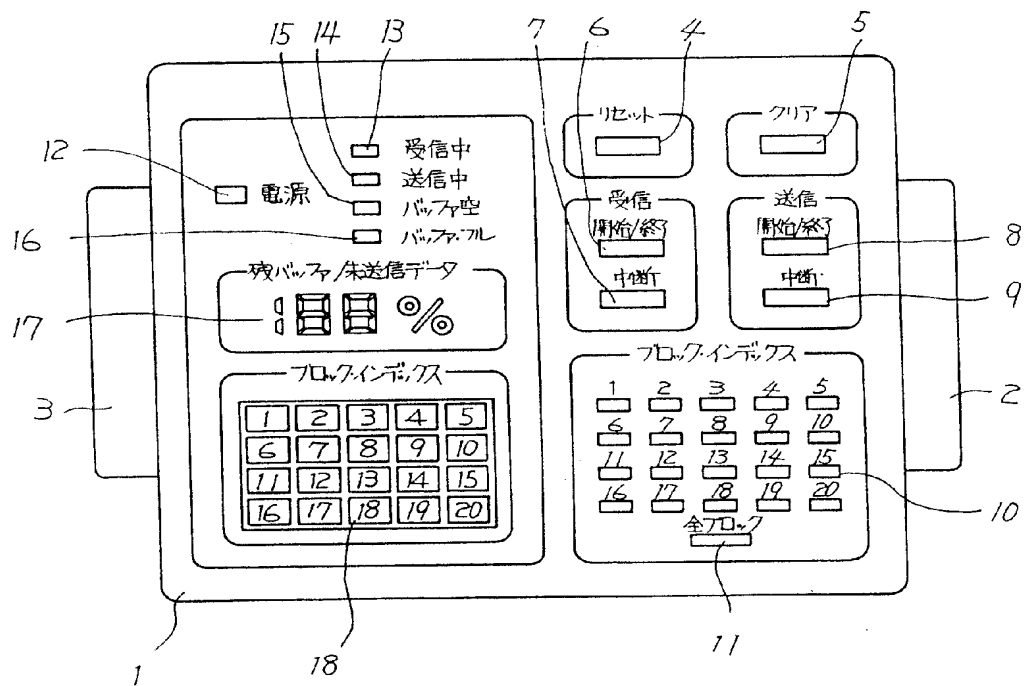
- 1 はプリンタ・バッファ本体、
- 2 は対プリンタ・ポート・コネクタ、
- 3 は対コンピュータ・ポート・コネクタ、
- 4 はリセット・ボタン、
- 5 はクリア・ボタン、
- 6 は受信開始／終了ボタン、

- 7 は受信中断ボタン、
- 8 は送信開始／終了ボタン、
- 9 は送信中断ボタン、
- 10 はブロック・インデックス・ボタン、
- 11 は全ブロック選択ボタン、
- 12 は電源インジケータ・ランプ、
- 13 は受信中表示ランプ、
- 14 は送信中表示ランプ、
- 15 はバッファ空表示ランプ、
- 16 はバッファ・フル表示ランプ、
- 17 は残バッファ／未送信データ容量表示ランプ、
- 18 はブロック・インデックス表示ランプ、
- 19 は電源スイッチ、
- 21 は電源部、
- 22 は CPU、
- 26 は対プリンタ・ポート、
- 27 は対コンピュータ・ポート、
- 29 は入力パネル、
- 31 は表示パネル、
- 33 は主メモリである。

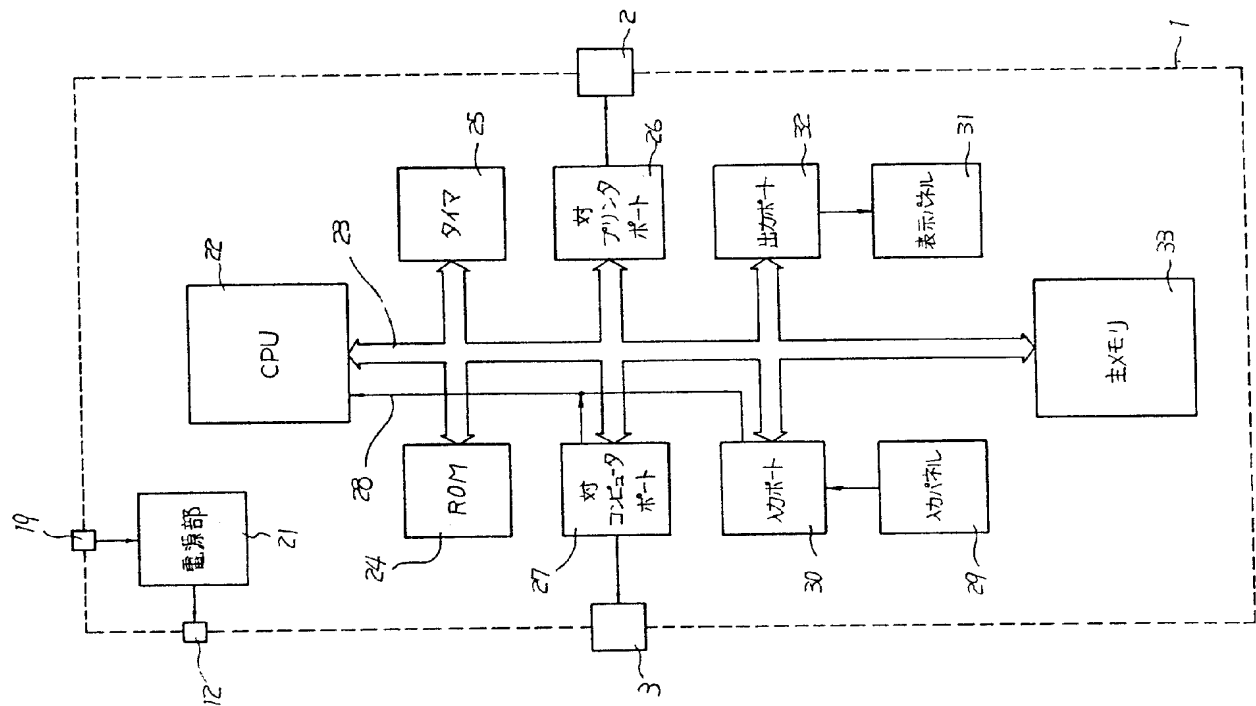
第 1 図



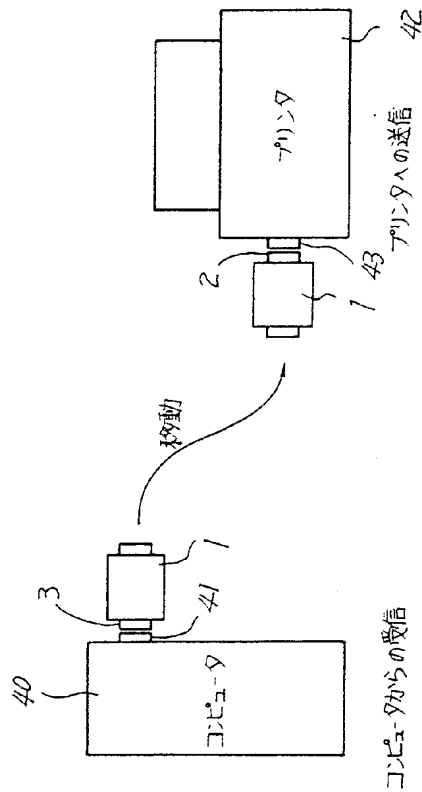
第2図



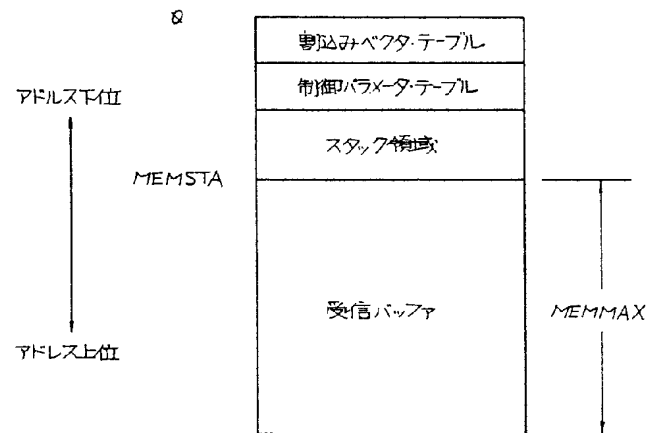
第3図



第4図



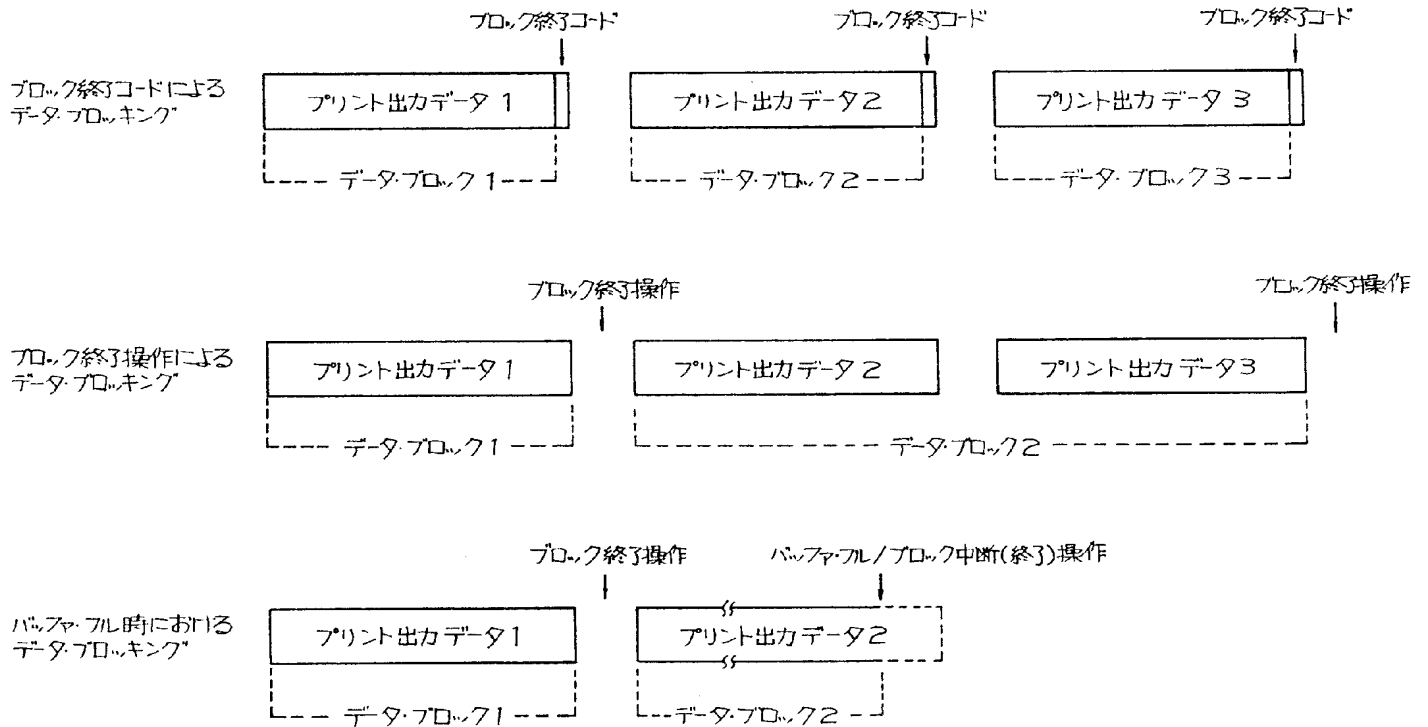
第7図



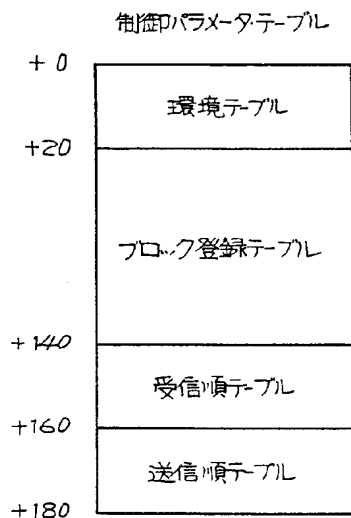
第5図

一般的な信号名	対コンピュータ・ポート			対プリンタ・ポート		
	信号名	入/出力	備考	信号名	入/出力	備考
-STROBE	-STROBE-C	入	非接続時"H"にプルアップ	-STROBE-P	出	
DATA0 ~ DATA 7	DATA0_C ~ DATA7_C	入	同上	DATA0_P ~ DATA7_P	出	
-ACKNLG	-ACKNLG-C	出		-ACKNLG-P	入	非接続時"H"にプルアップ
BUSY	BUSY-C	出		BUSY-P	入	同上
PE	PE-C	出		PE-P	入	同上
SLCT	SLCT-C	出	"H"固定出力	SLCT-P	入	同上
-ERROR	-ERROR-C	出		-ERROR-P	入	同上
-INIT	-INIT-C	入	"L"検出時にリセット	-INIT-P	出	送信動作開始時に出力
-AUTOFEEDXT	-AUTOFEEDXT-C	入	インシャイス時にモニタ/セーブ	-AUTOFEEDXT-P	出	同上
-SLCTIN	-SLCTIN-C	入	同上	-SLCTIN-P	出	同上

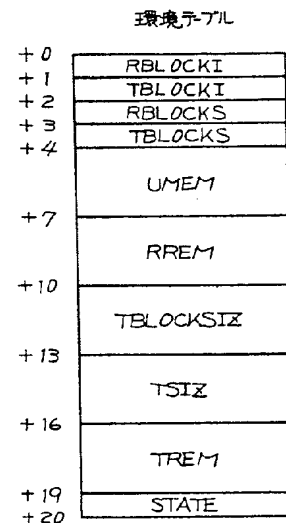
第6図



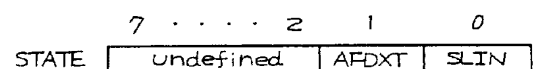
第8図



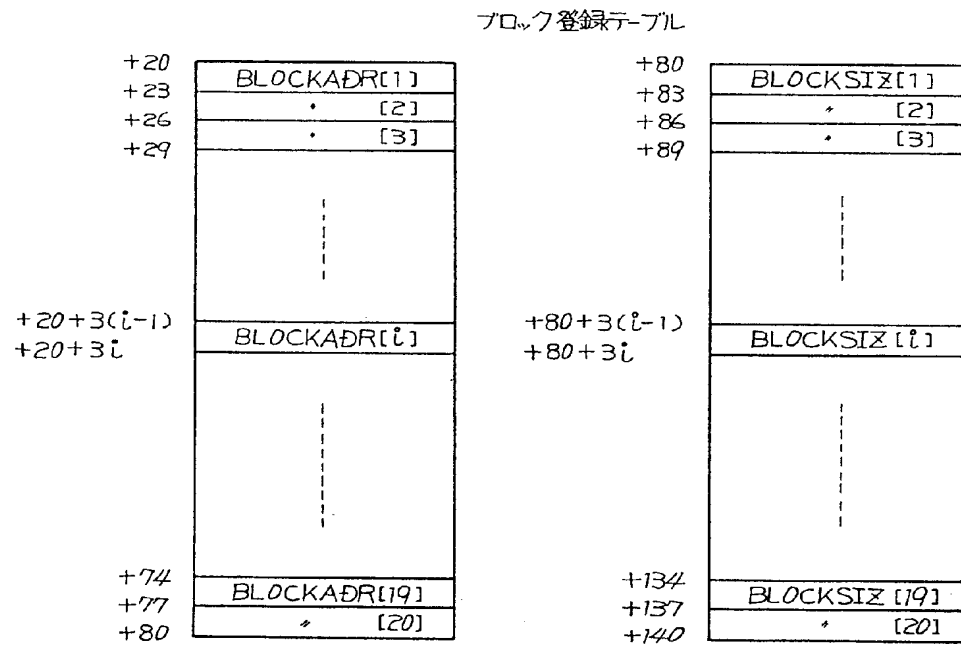
第9図



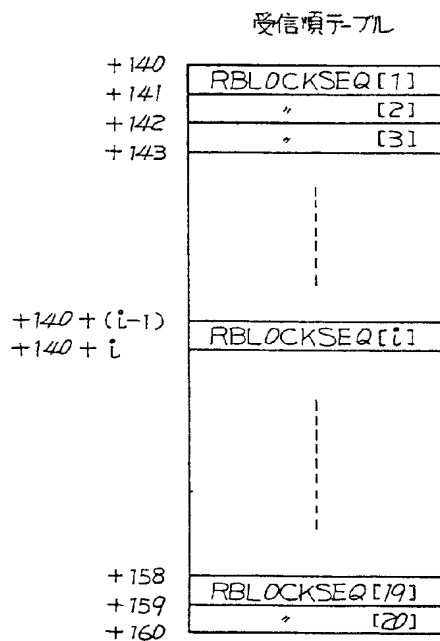
第10図



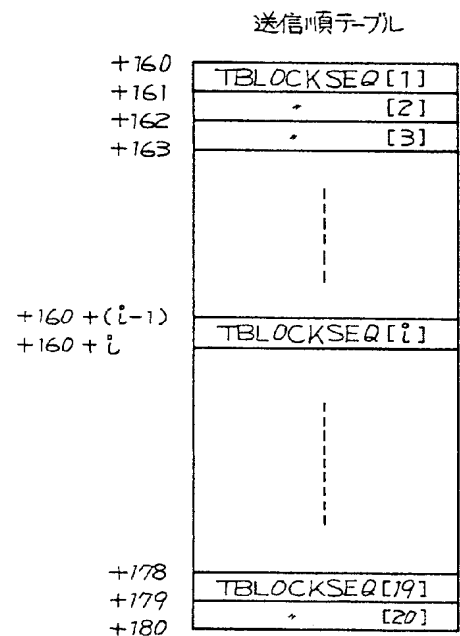
第 11 図



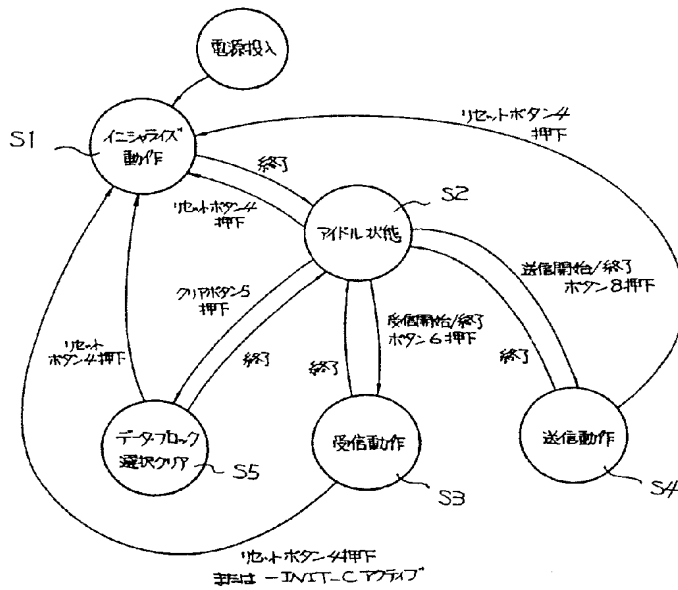
第 12 図



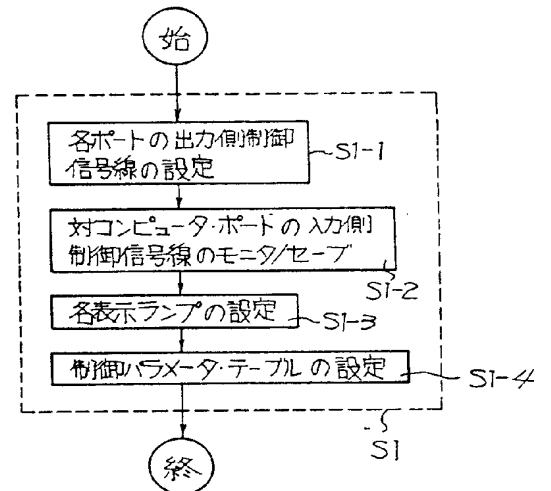
第 13 図



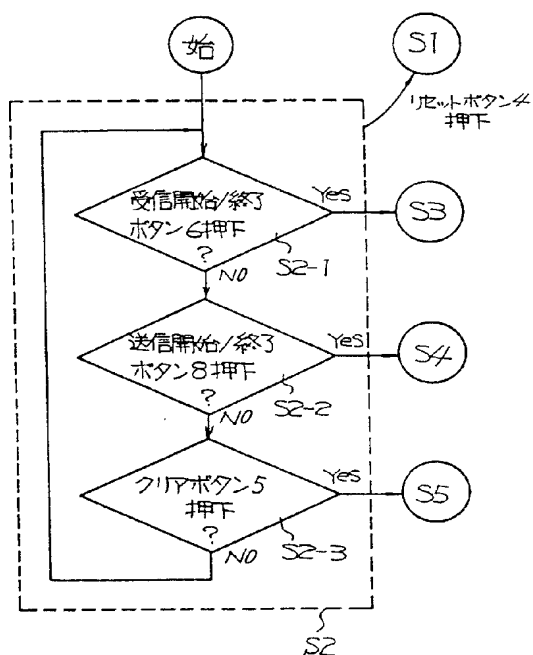
第 14 图



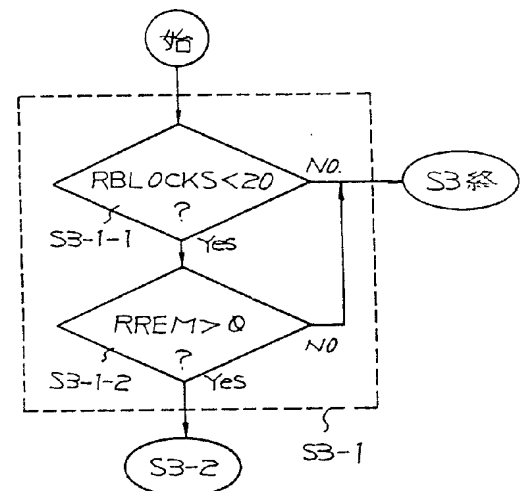
第 15 回



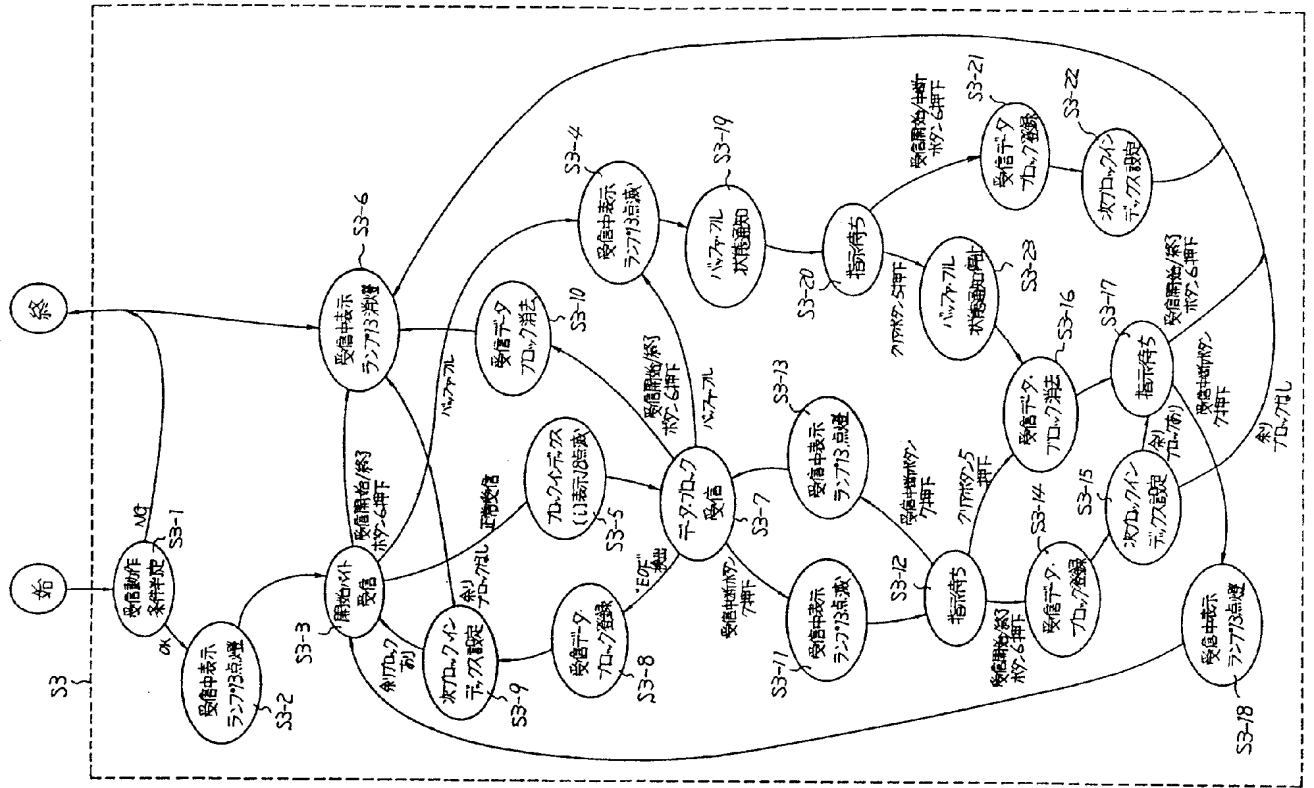
第 16 义



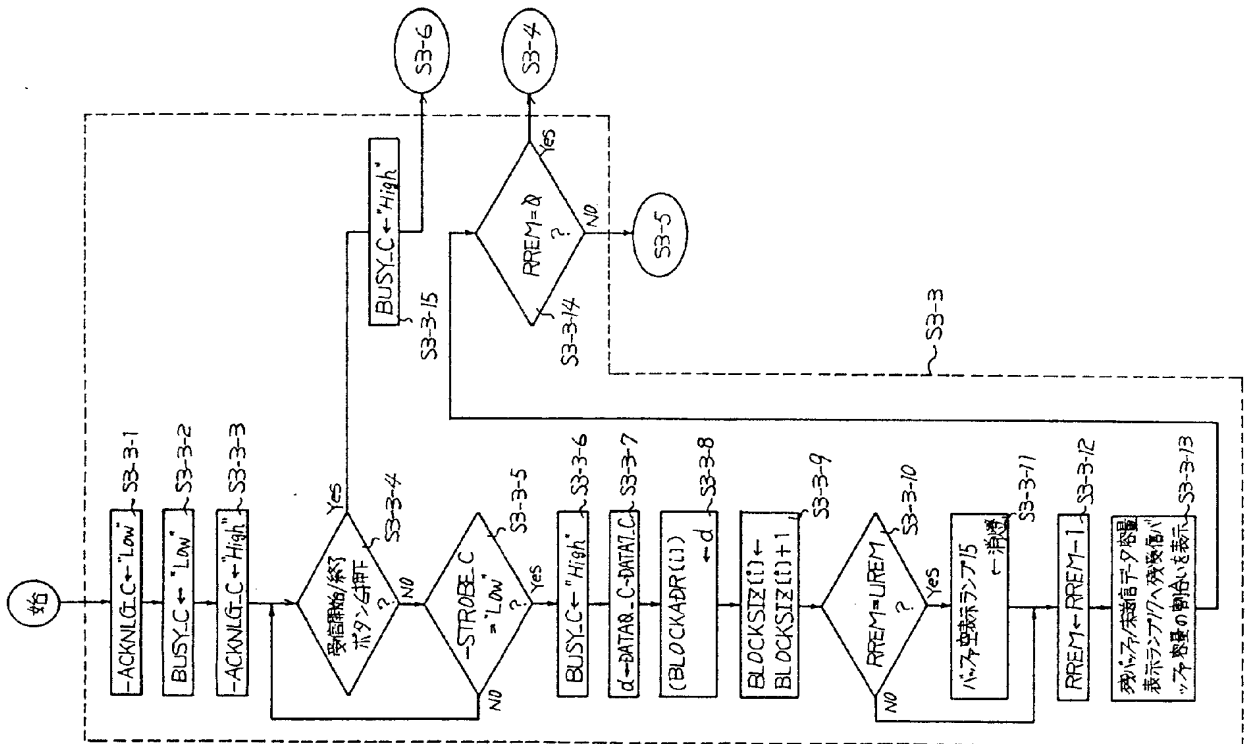
第 18 回



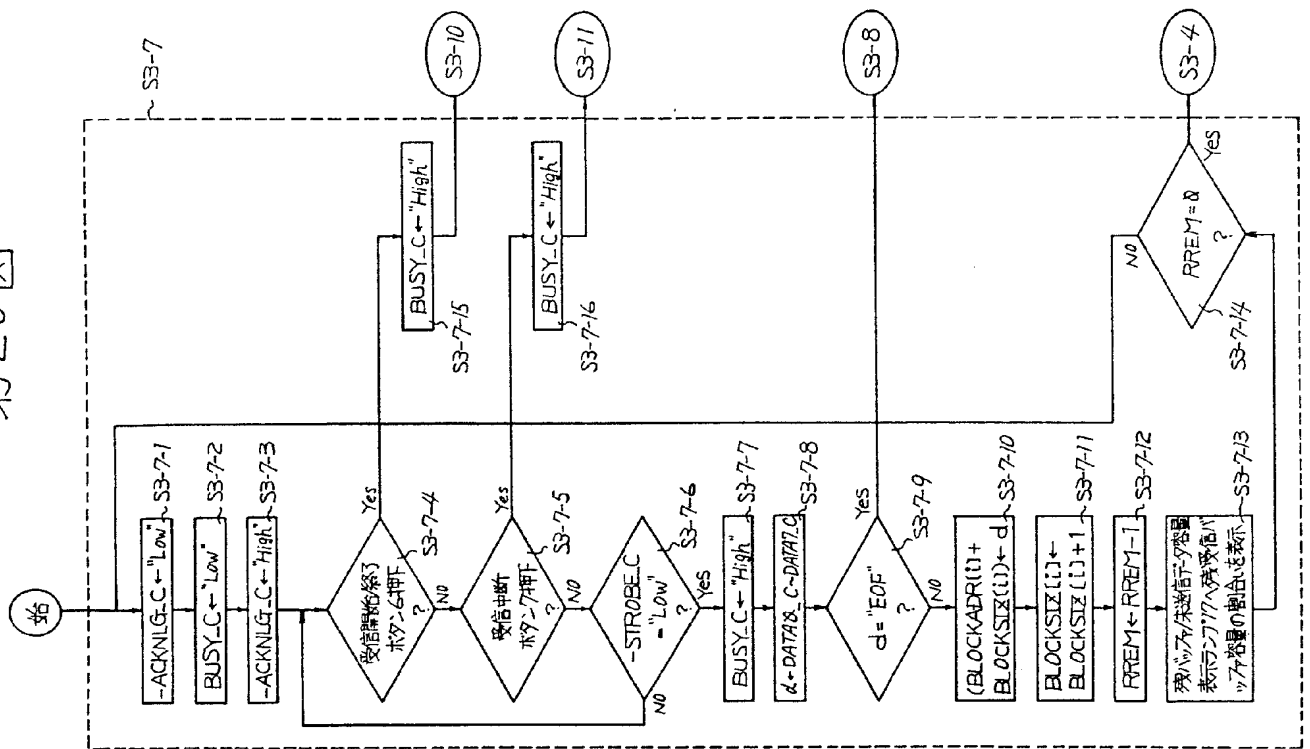
第17図



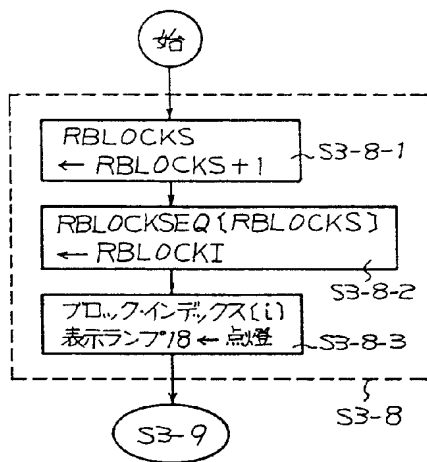
第19図



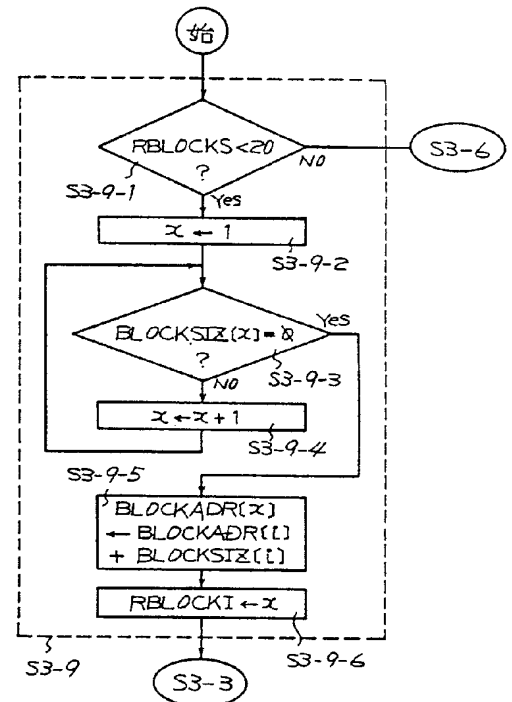
第20図



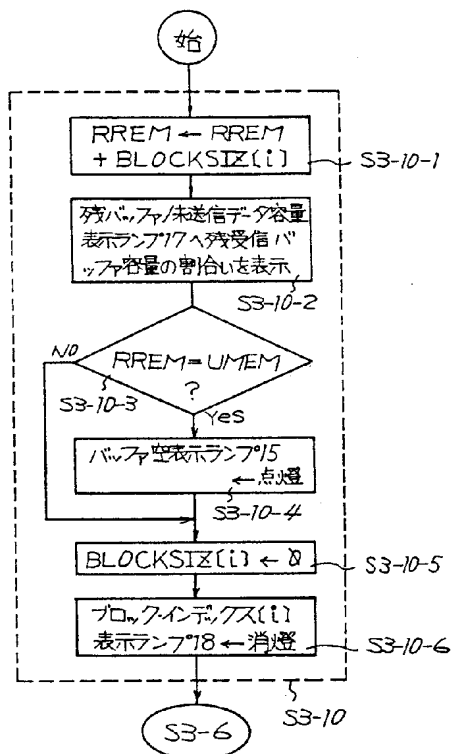
第21図



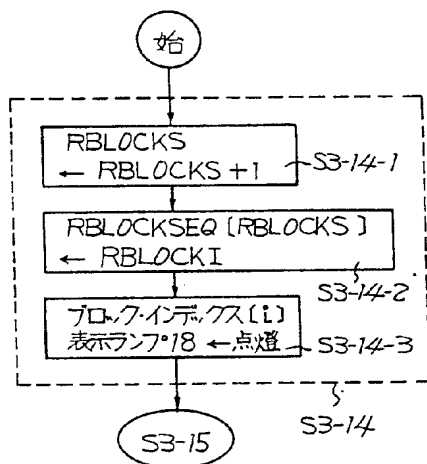
第22図



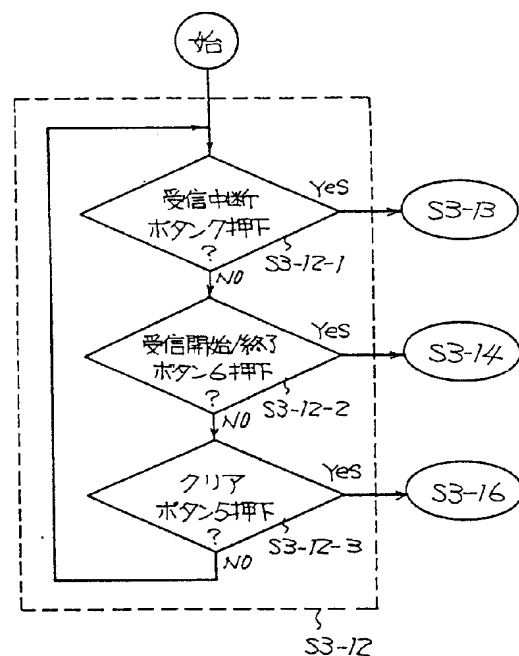
第23図



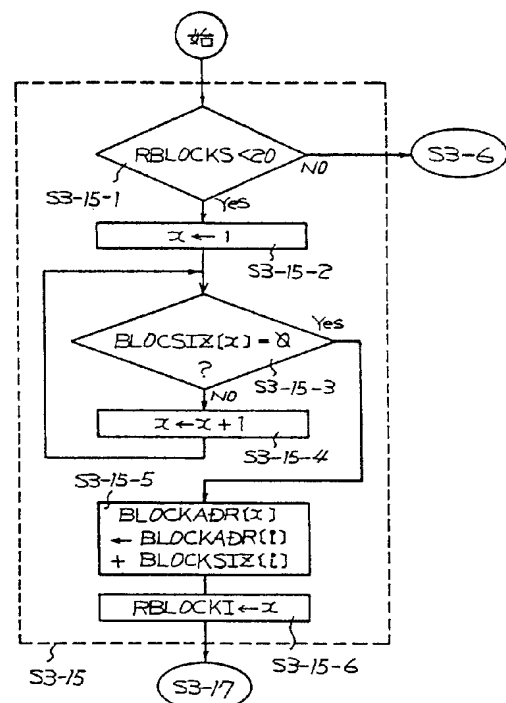
第25図



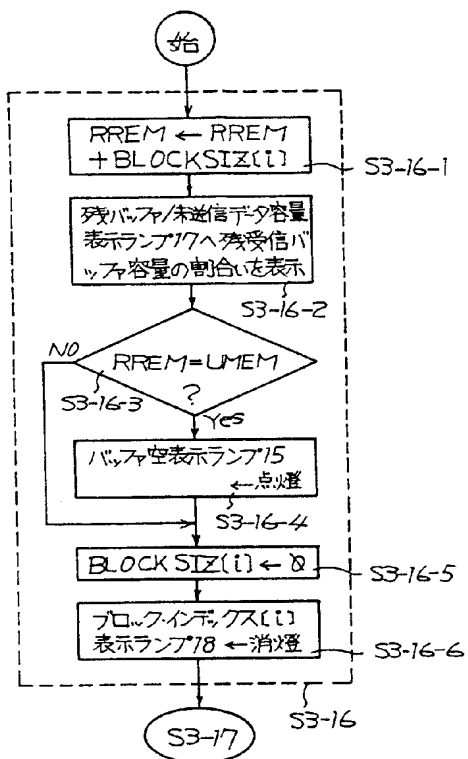
第24図



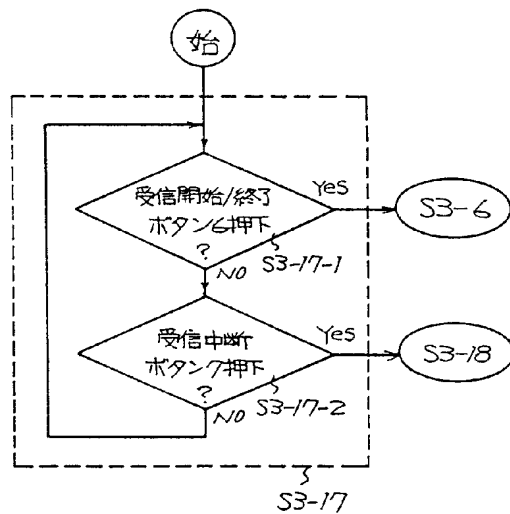
第26図



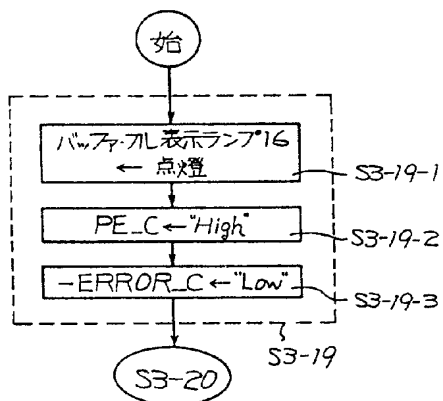
第27図



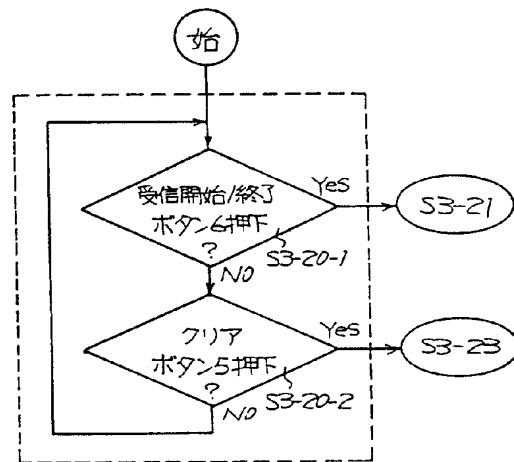
第28図



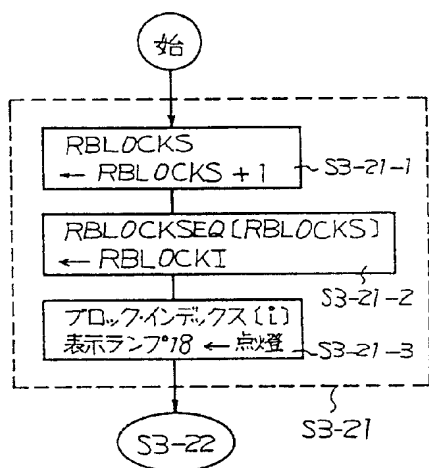
第29図



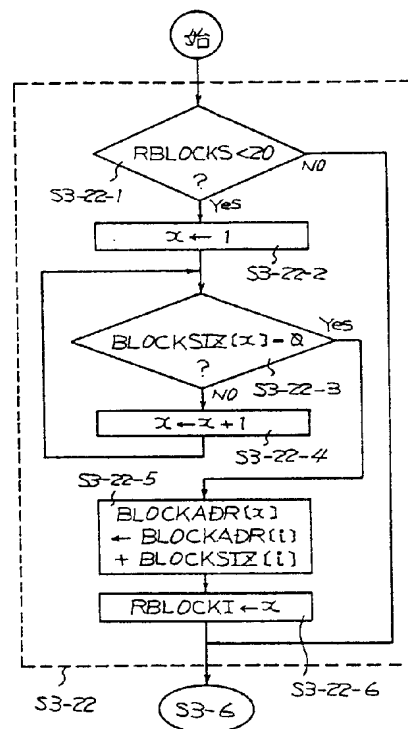
第30図



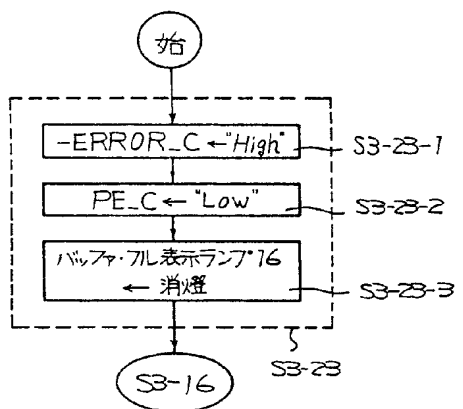
第31図



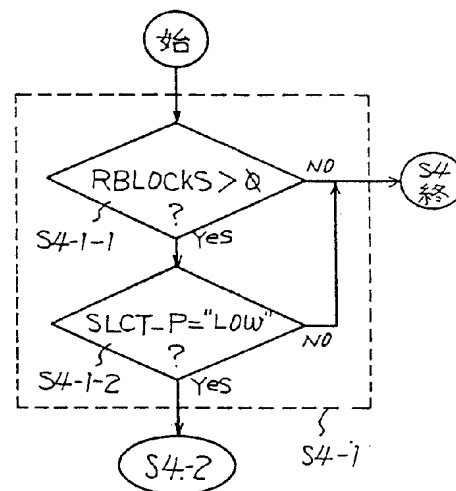
第32図



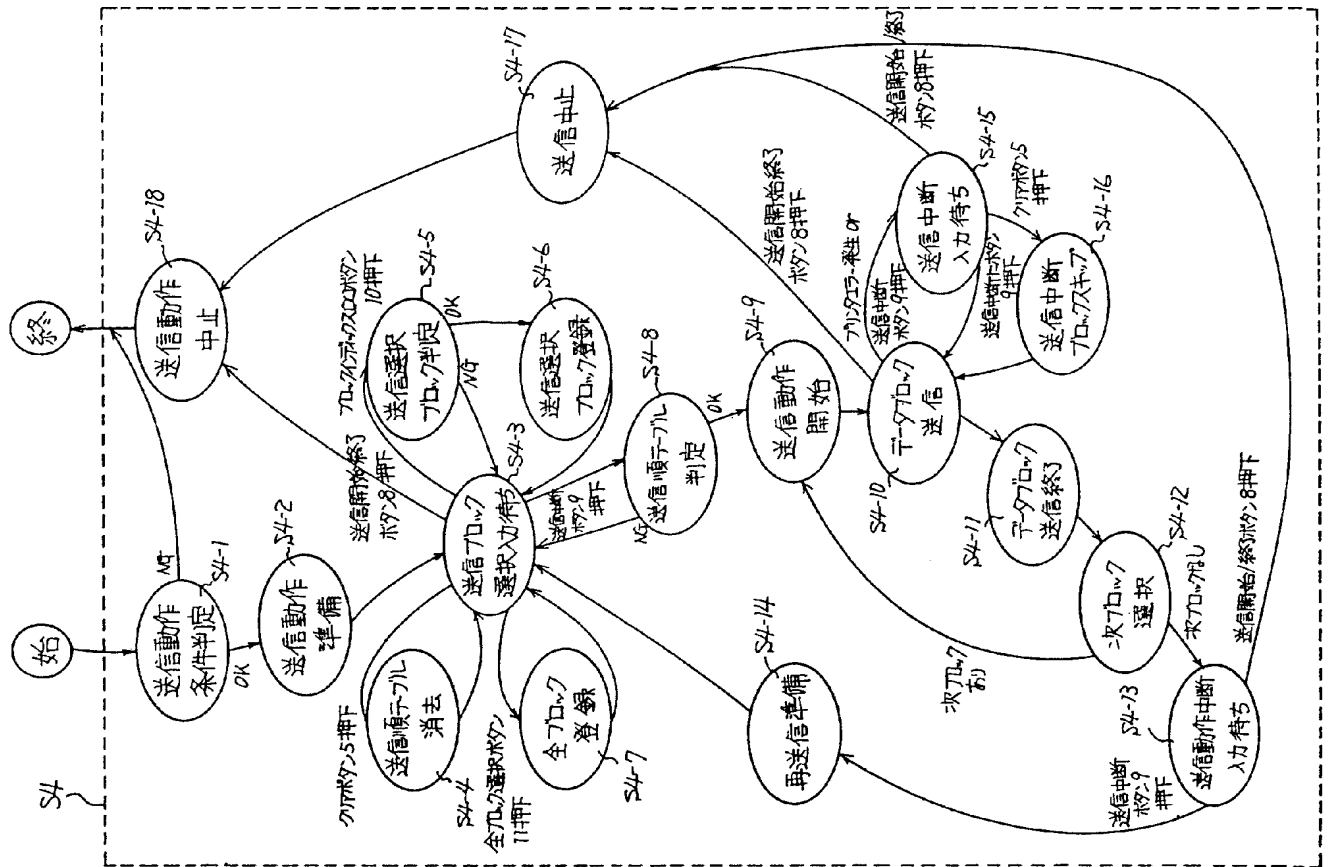
第33図



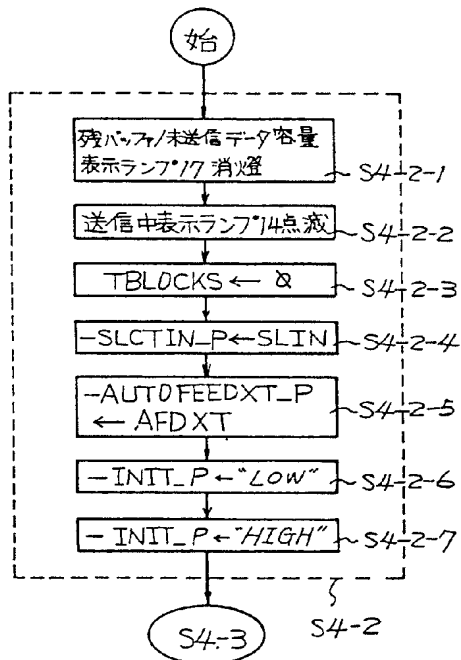
第35図



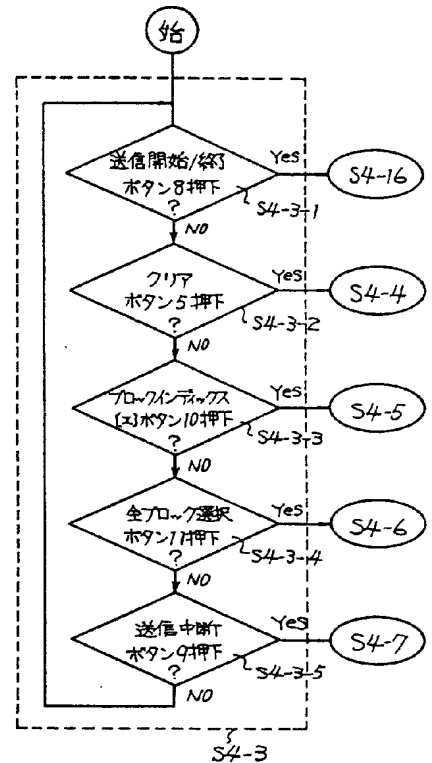
第34図



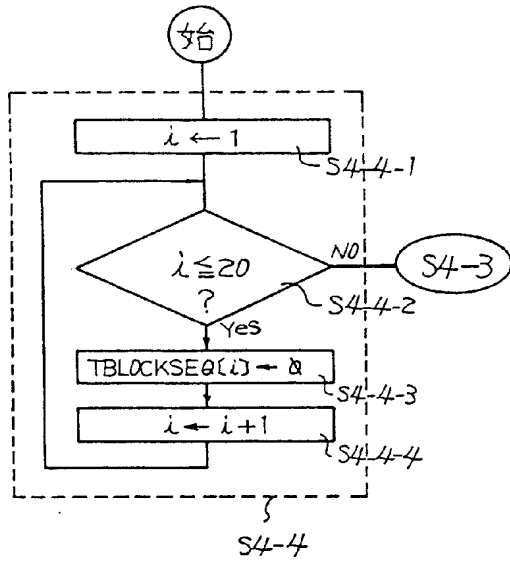
第36図



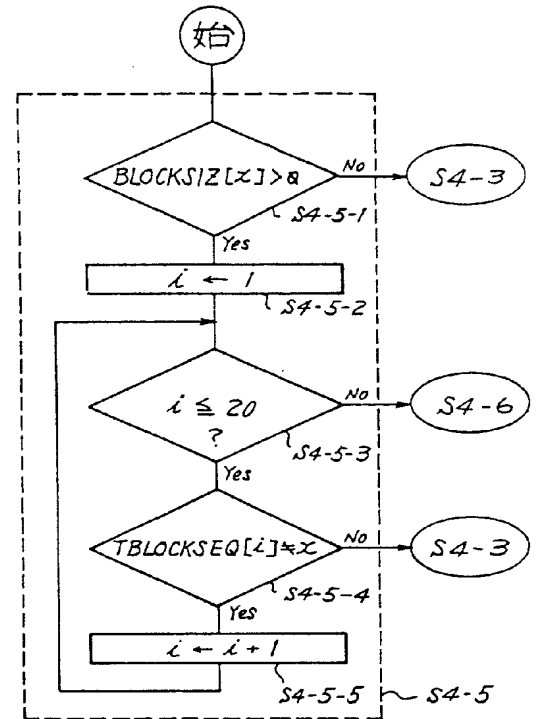
第37図



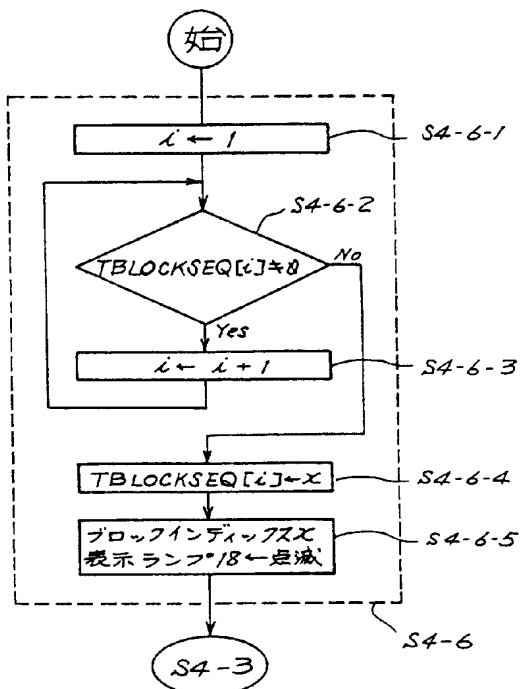
第38図



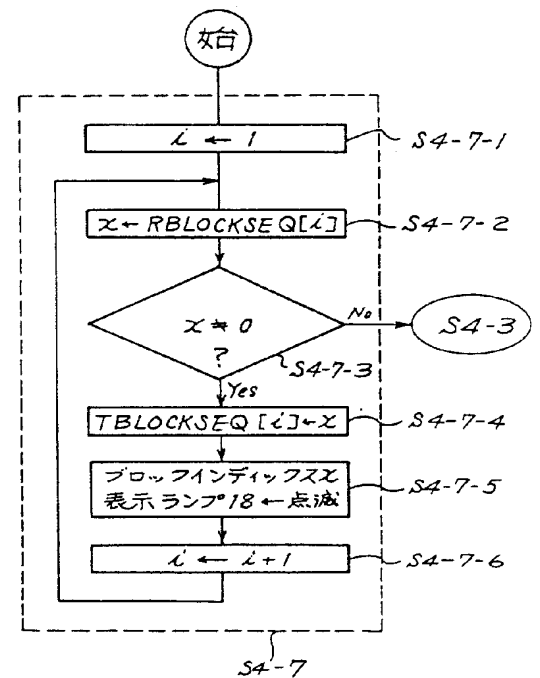
第39図



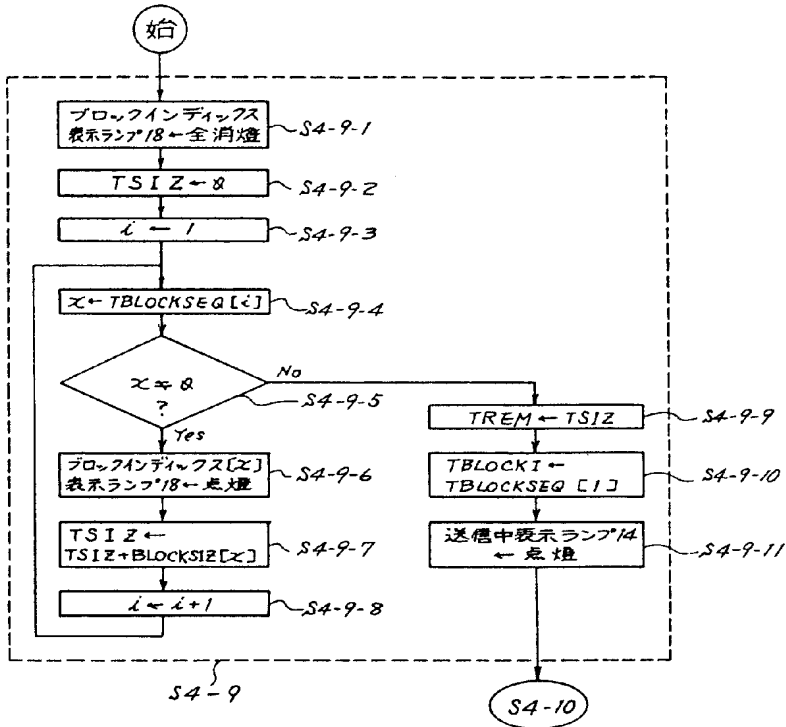
第40図



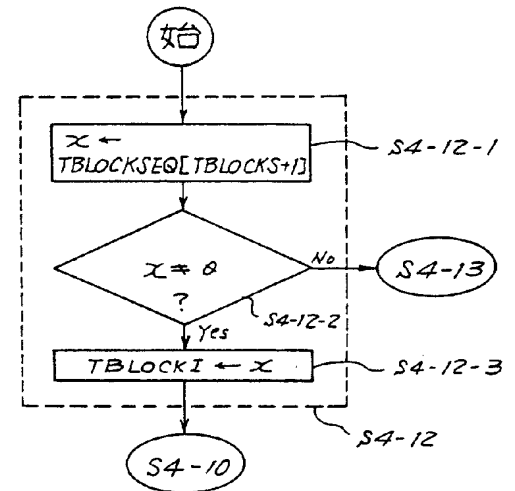
第41図



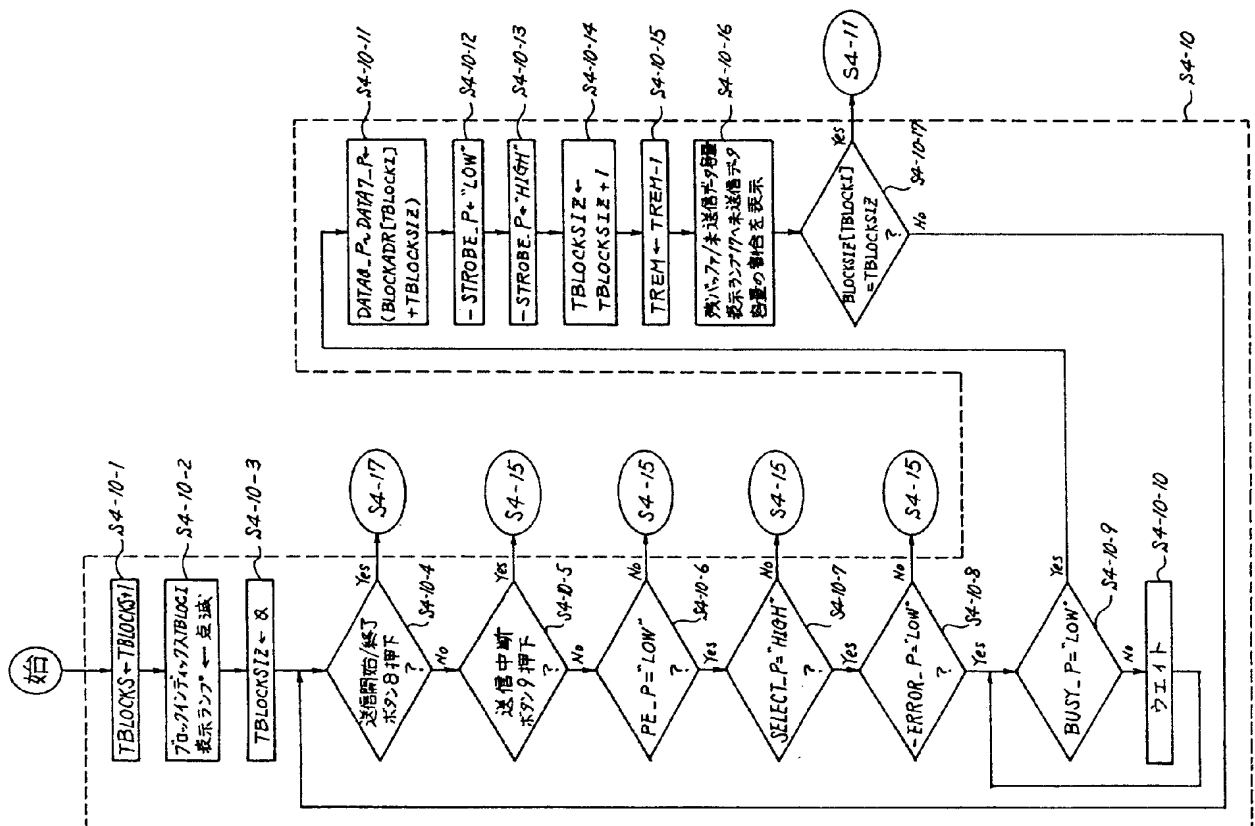
第 42 図



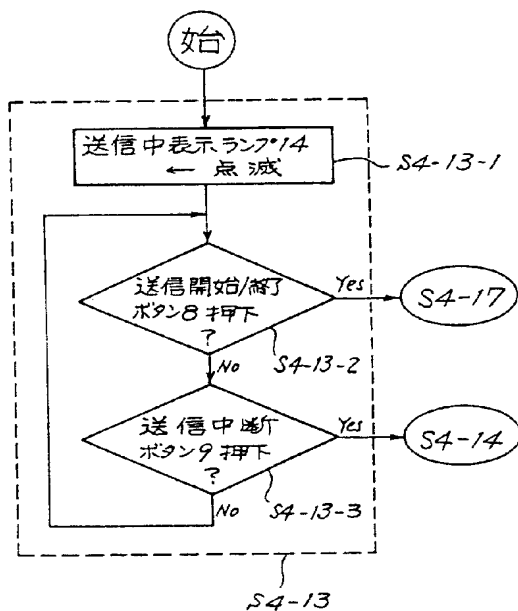
第 44 図



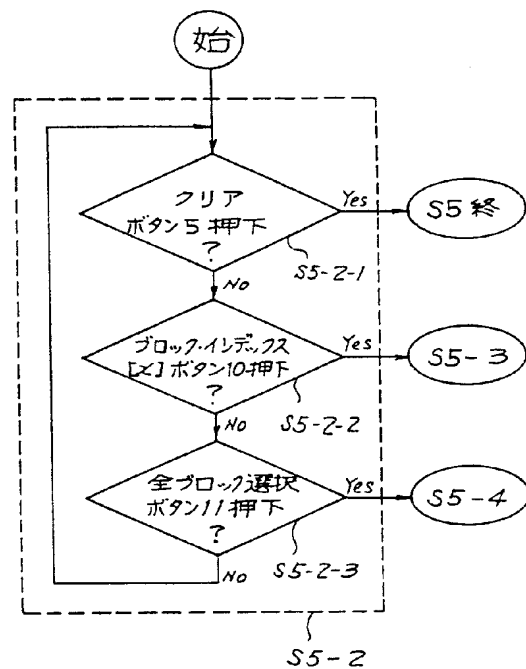
第 43 圖



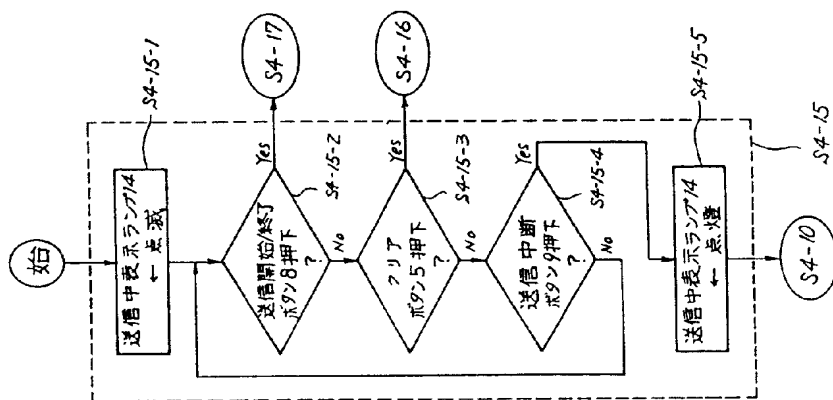
第 45 図



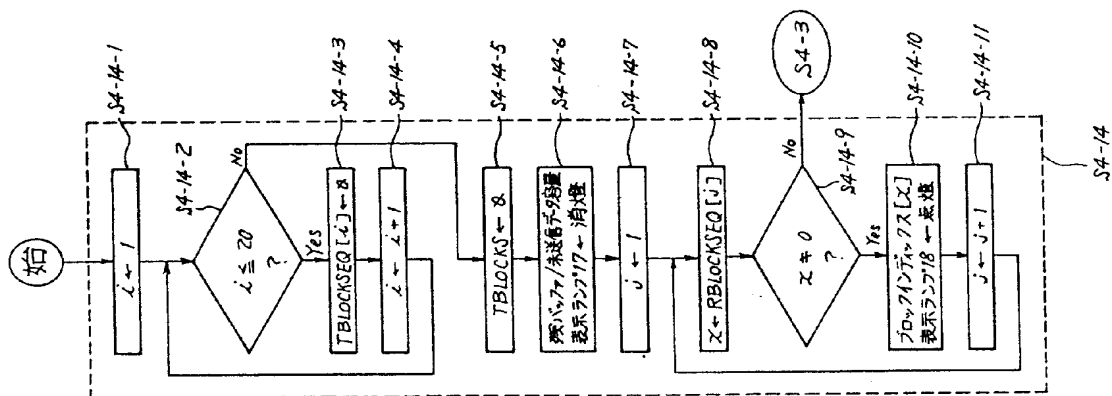
第 50 図



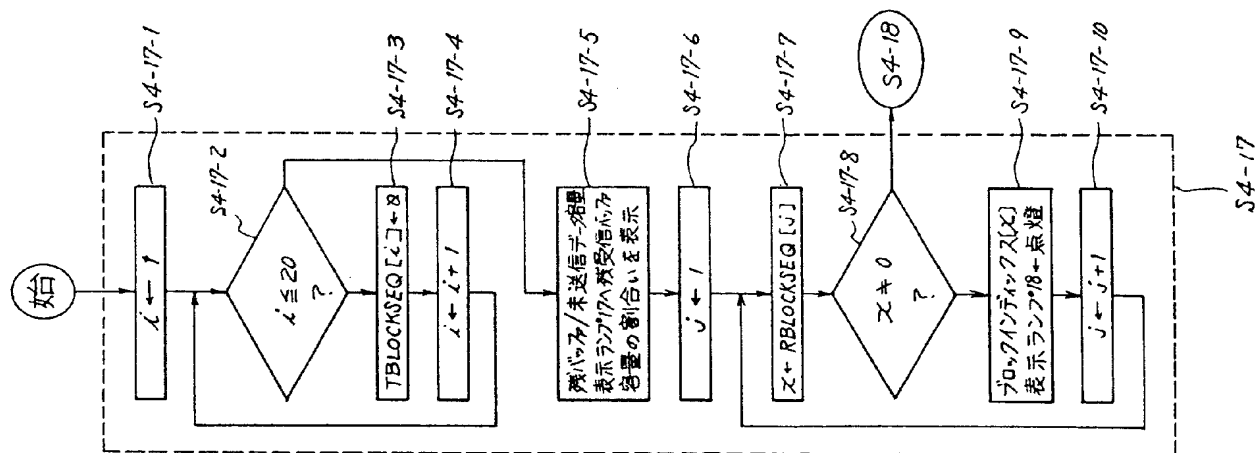
第 47 図



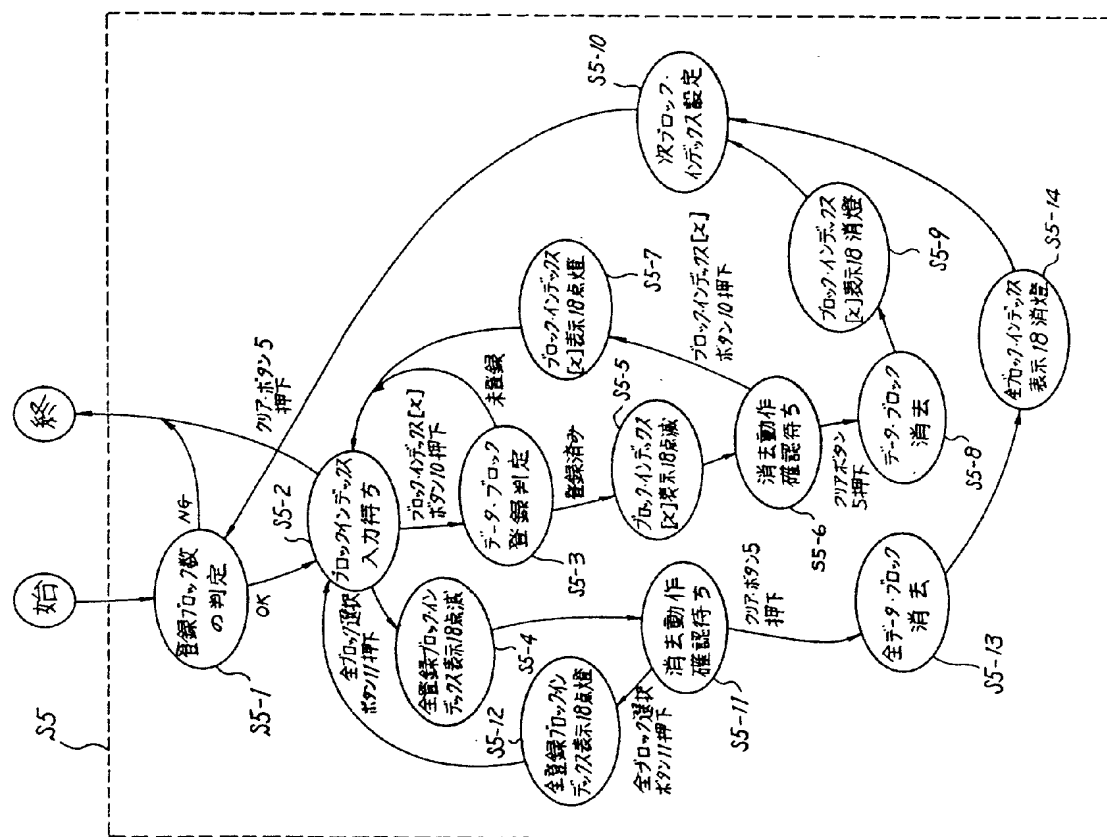
第 46 図



第 48 圖

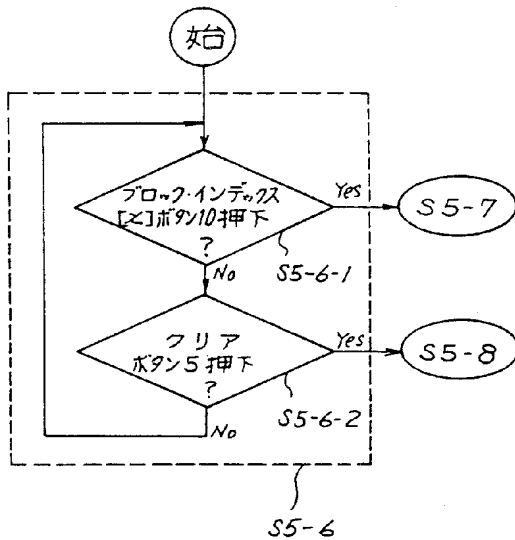


第 49 圖

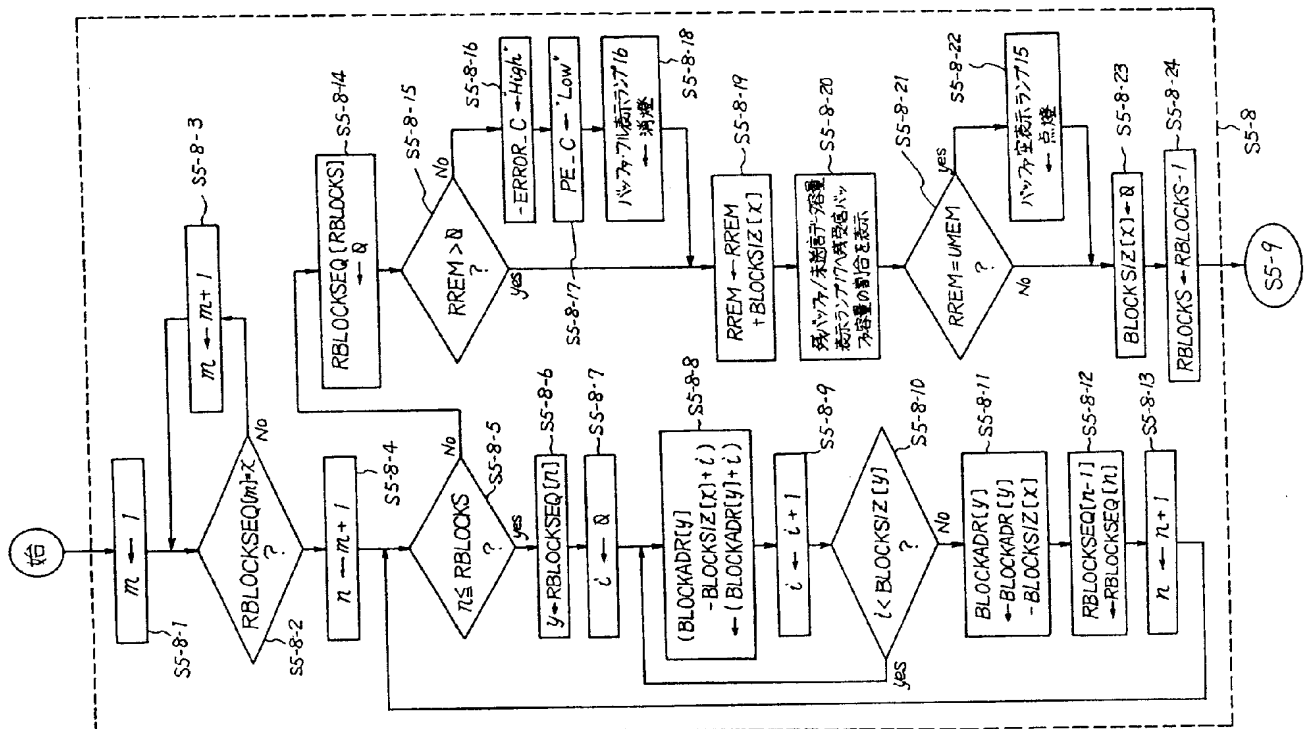


第 53 ☒

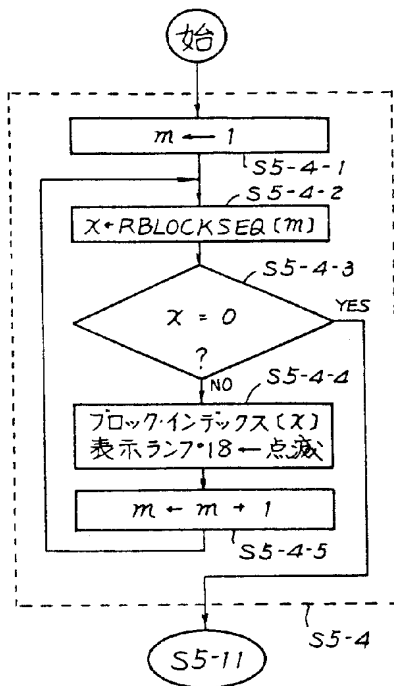
第 51 回



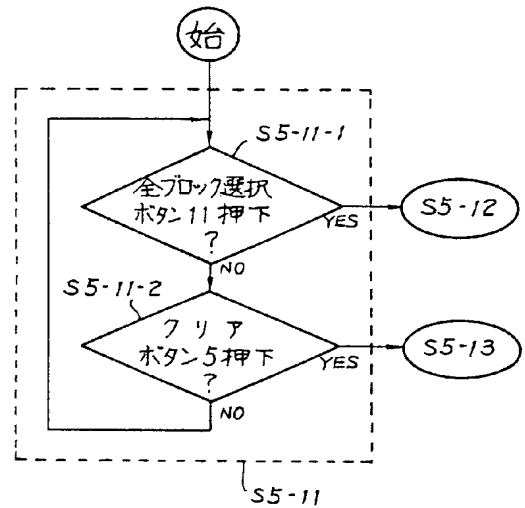
第 52 页



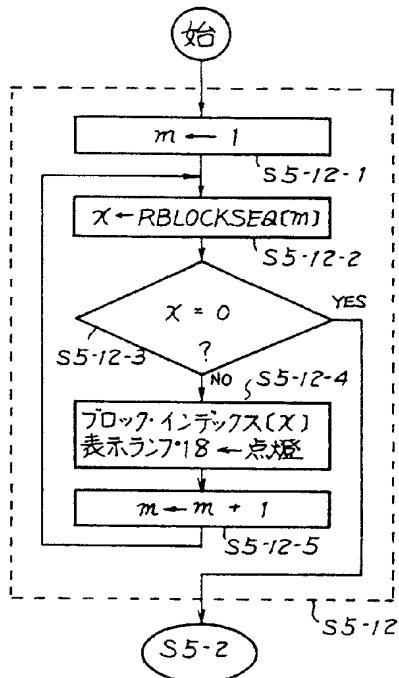
第54図



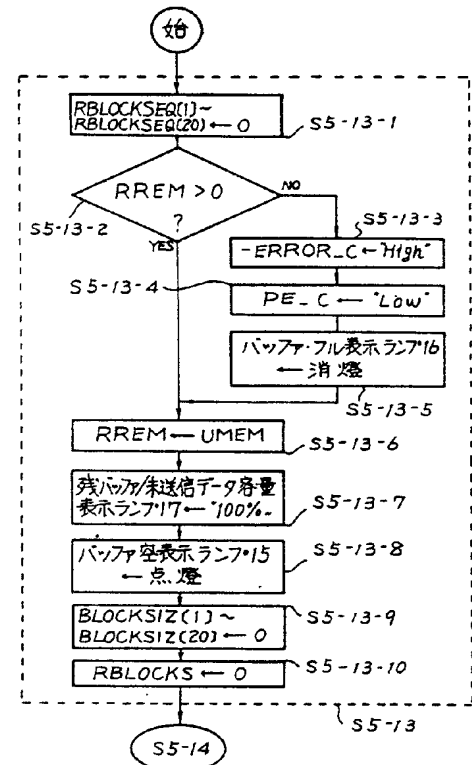
第55図



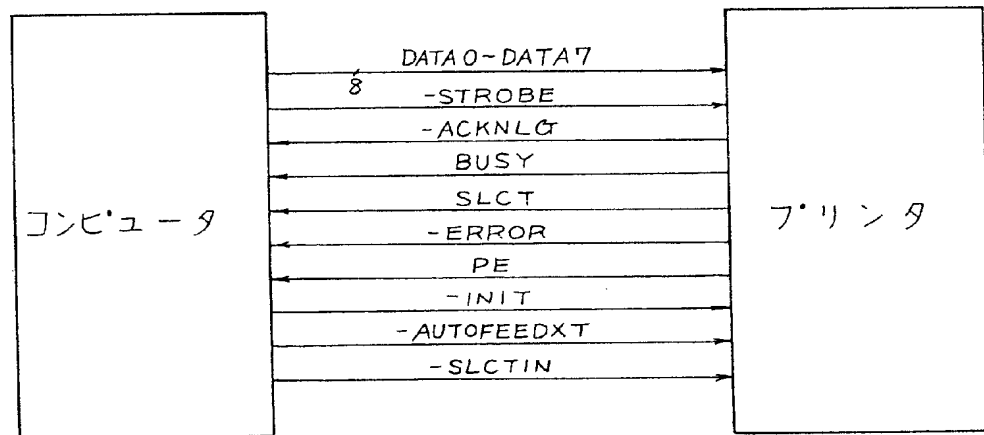
第56図



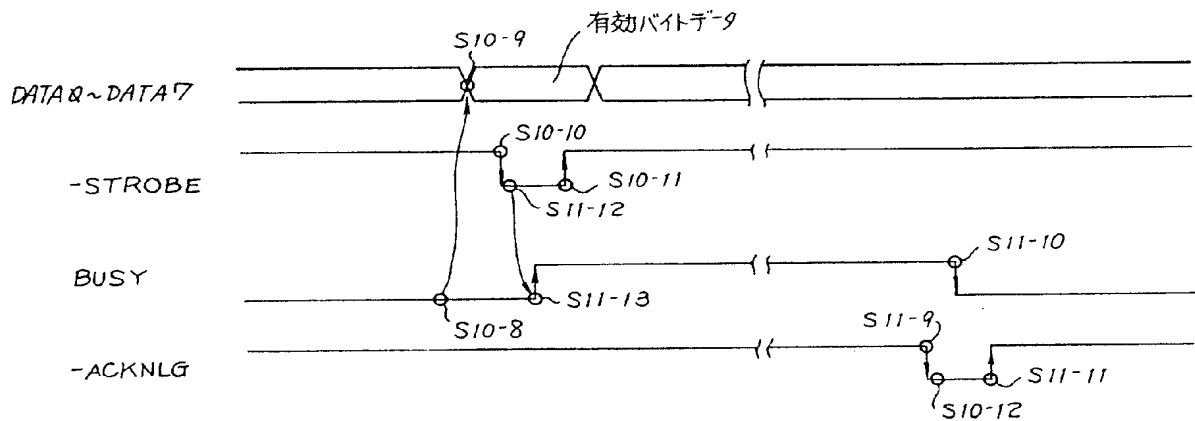
第57図



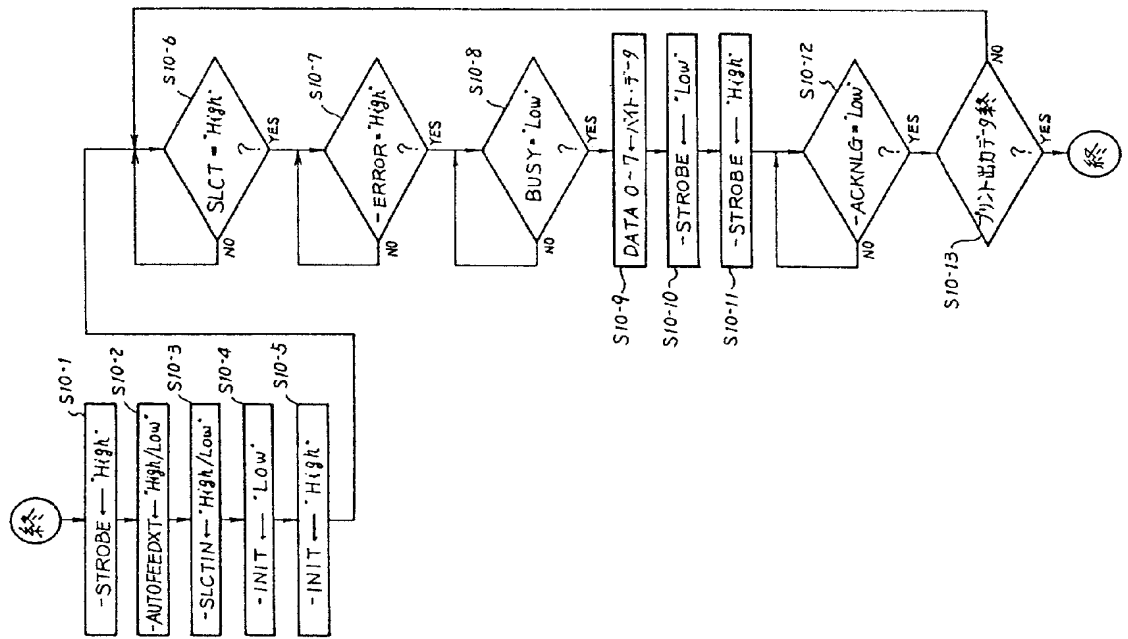
第58図



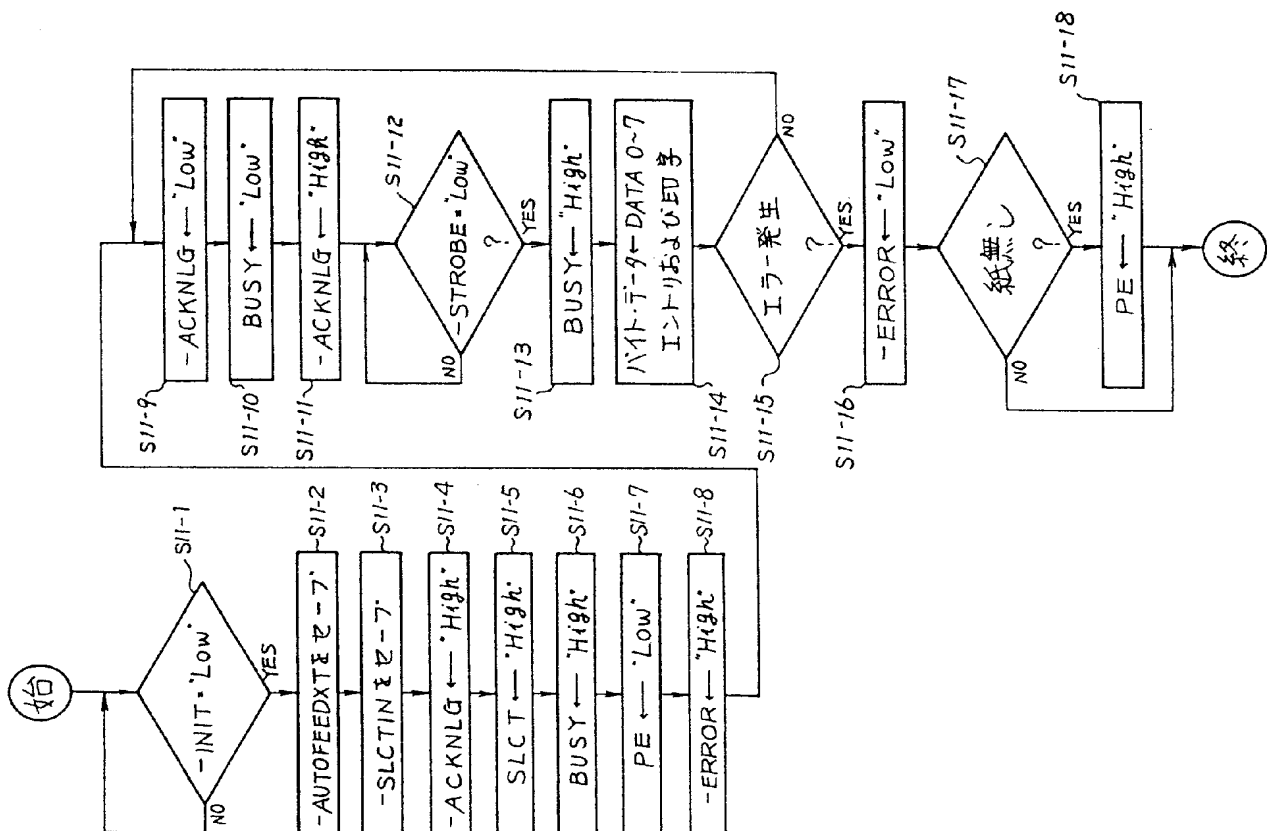
第59図



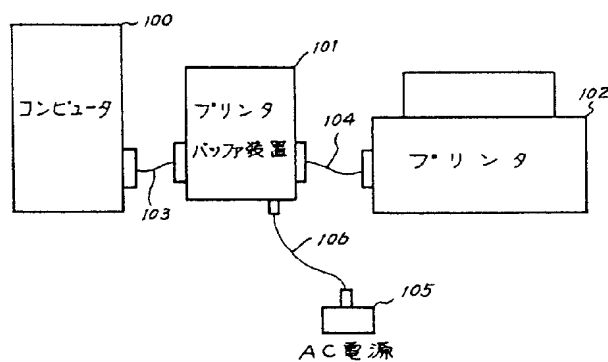
第60図



第61図



第62図



第63図

